2003

Arithmetic logic UNIT (ALU) design using reconfigurable CMOS logic

Chandra Srinivasan
Louisiana State University and Agricultural and Mechanical College, csrini2@lsu.edu

ARITHMETIC LOGIC UNIT (ALU) DESIGN USING RECONFIGURABLE CMOS LOGIC

A Thesis

Submitted to the Graduate Faculty of the
Louisiana State University and
Agricultural and Mechanical College
in partial fulfillment of the
requirements for the degree of
Master of Science in Electrical Engineering

in
The Department of Electrical and Computer Engineering

by
Chandra Srinivasan
Bachelor of Engineering, Mysore University, 1997
December 2003

To
My parents
and
in loving memory of my grandmother

Acknowledgements

I would like to acknowledge certain people who have encouraged, supported and helped me complete my thesis at LSU.

I am very grateful to my advisor Dr. A. Srivastava for his guidance, patience and understanding throughout this work. His suggestions, discussions and constant encouragement has helped me to get a deep insight in the field of VLSI design. I would like to thank Dr. M. Feldman and Dr S. Kak for sparing their time to be a part of my thesis advisory committee. I am very thankful to Electrical Engineering Department for supporting me financially during my stay at LSU. I take this opportunity to thank my friends Harish, Kavitha, Sajida, Sunitha and Anand for their help and encouragement. I would also like to thank all my friends here who made my stay at LSU an enjoyable and a memorable one. I extend a special thank-you to Srinivas, my constant companion and beloved friend. Last of all I thank the Almighty Lord for keeping me in good health and spirits throughout my stay at LSU.

Table of Contents

List of Tables

List of Figures

Abstract

Using the reconfigurable logic of multi-input floating gate MOSFETs, a 4-bit ALU has been designed for 3V operation. The ALU can perform four arithmetic and four logical operations**.**

Multi-input floating gate (MIFG) transistors have been promising in realizing increased functionality on a chip. A multi-input floating gate MOS transistor accepts multiple inputs signals, calculates the weighted sum of all input signals and then controls the ON and OFF states of the transistor. This enhances the transistor function to more than just switching. This changes the way a logic function can be realized. Implementing a design using multi-input floating gate MOSFETs brings about reduction in transistor count and number of interconnections. The advantage of bringing down the number of devices is that a design becomes area efficient and power consumption reduces. There are several applications that stress on smaller chip area and reduced power. Multi-input floating gate devices have their use in memories, analog and digital circuits.

In the present work we have shown successful implementation of multi-input floating gate MOSFETs in ALU design. A comparison has been made between adders using different design methods w.r.t transistor count. It is seen that our design, implemented using multi-input floating gate MOSFETs, uses the least number of transistors when compared to other designs. The design was fabricated using double polysilicon standard CMOS process by MOSIS in 1.5μm technology. The experimental waveforms and delay measurements have also been presented.

Chapter 1

Introduction

Advancement in VLSI technology has allowed following Moore's law [1] for doubling component density on a silicon chip after every three years. Though MOS transistors have been scaled down, increased interconnections have limited circuit density on a chip. Furthermore, the size of transistor is limited by hot-carrier phenomena and increase in electric field that lead to degradation of device performance and device lifetime [2]. It has become essential to look into other methods of adding more functionality to a MOS transistor, such as, the multiple-input floating gate MOS transistor structure proposed by Shibata and Ohmi [3]. An enhancement in the basic function of a transistor has, thus, allowed for designs to be implemented using fewer transistors and reduced interconnections. In published literature [4-7], many integrated circuits have been reported which are using multi-input floating gate MOSFETs in standard CMOS process.

The arithmetic logic unit (ALU) is the core of a CPU in a computer. The adder cell is the elementary unit of an ALU. The constraints the adder has to satisfy are area, power and speed requirements. Some of the conventional types of adders are ripple-carry adder, carry-lookahead adder, carry-skip adder and Manchester carry chain adder [8]. The delay in an adder is dominated by the carry chain. Carry chain analysis must consider transistor and wiring delays.

Ripple carry adder is an n-bit adder built from full adders. Figure 1.1 shows a 4-bit ripple carry adder. Carry-lookahead adders first compute carry propagate and generate and then computes SUM and CARRY from them. It allows for carry to be computed in

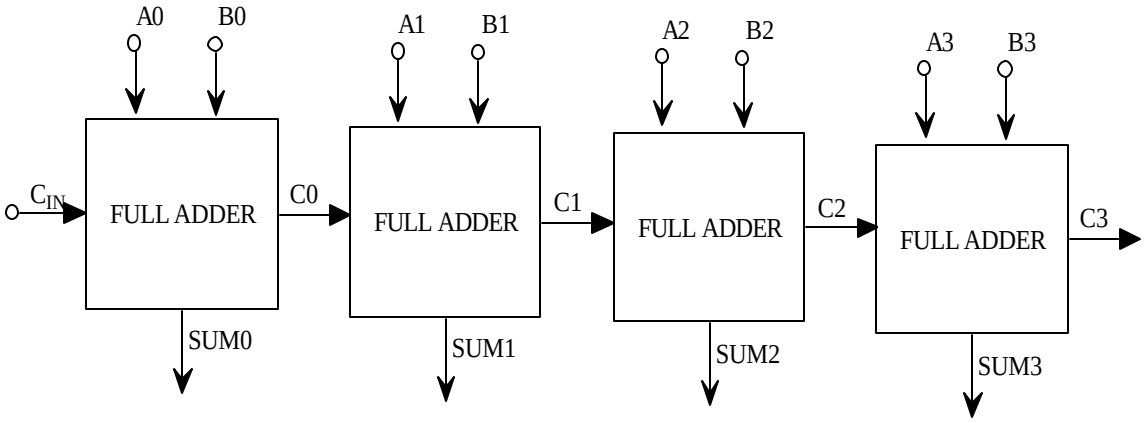


Figure 1.1: Block diagram of a 4-bit ripple carry adder (RCA).

each bit. Figure 1.2 shows a 4-bit carry-lookahead adder. Carry-lookahead unit requires complex wiring between adders and lookahead unit, as the values must be routed back to adder from lookahead unit. Layout becomes complex with multiple levels of lookahead. Figure 1.3 shows a 4-bit carry-skip adder and skip module used. The skip module determines whether it could just pass a carry in (C_{IN}) the next four bits for addition or it has to wait until the carry out (C_3) propagates through the last full adder in the design. In essence, the skip module can make the carry in (C_{IN}) appear to skip through the four full adders. The Manchester carry chain adder uses a precharged carry chain with P and G signals. Propagate signal P_i is the XOR of input bits A_i and B_i and generate signal G_i is the NAND of input bits A_i and B_i. Propagate signal connects adjacent carry bits and Generate signal discharge the carry bit. Figure 1.4 shows a Manchester carry chain. When input bits are '0', G_i is HIGH and hence the carry out node is discharged. When one of the input bits is '1', then P_i is HIGH and carry out follows carry in. When both bits are '1', then both G_i and P_i are LOW; hence carry out node remains isolated from carry in and ground. As the node is pre-charged to a HIGH state the carry out remains HIGH. Each of the adder configurations may or may not require additional logic apart from full adder design. Table 1.1 shows approximately how many additional gates and transistors are required for each of the adder configurations. In terms of area efficiency ripple carry adder is preferred. Keeping in mind small layout area and less number of interconnections our ALU has been designed using ripple carry configuration. However, the delay time for worst case is more when compared to other adders [8].

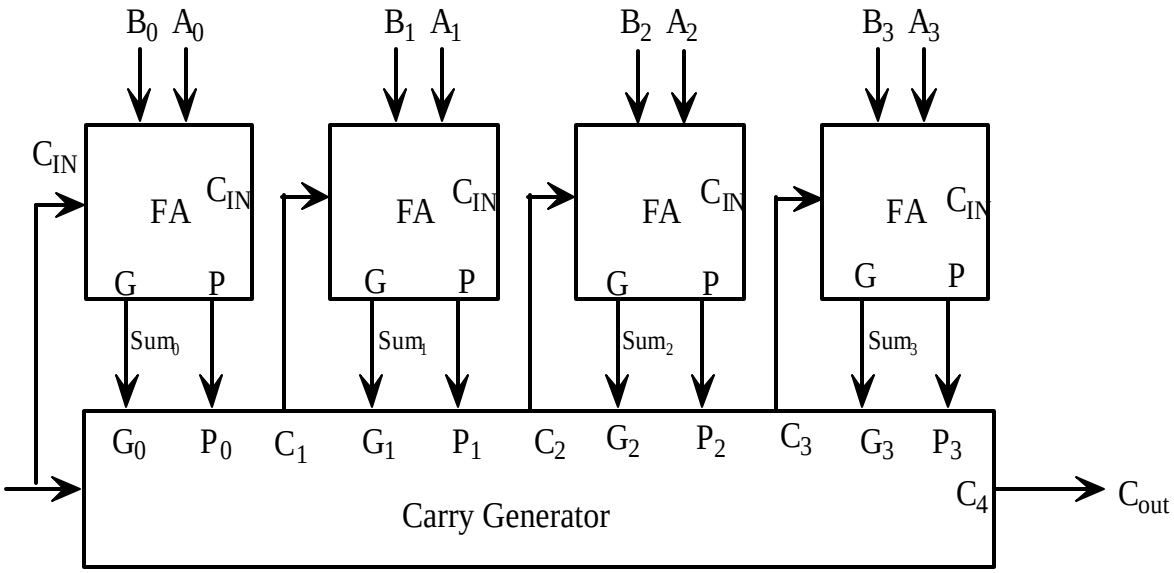


Figure 1.2: Block diagram of a 4-bit carry-lookahead adder (CLA).

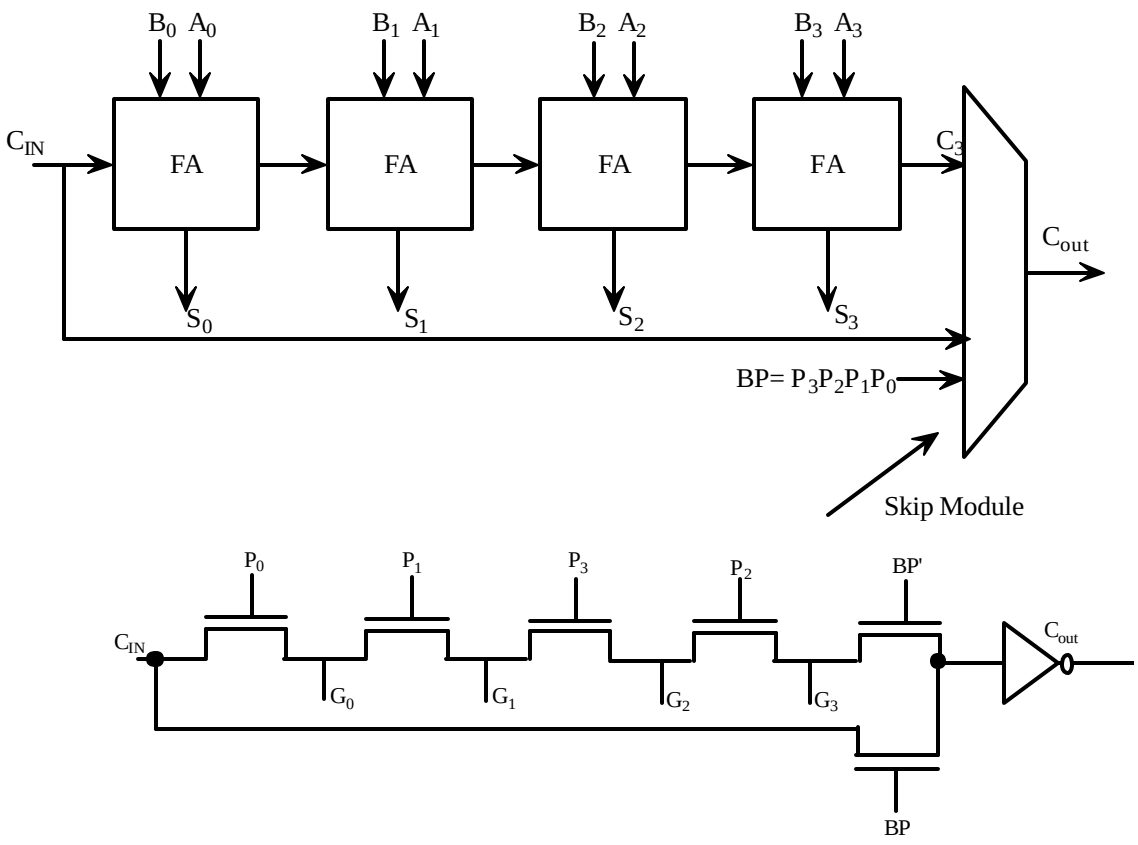


Figure 1.3: Block diagram of a 4-bit carry-skip adder (CSA) with skip module.

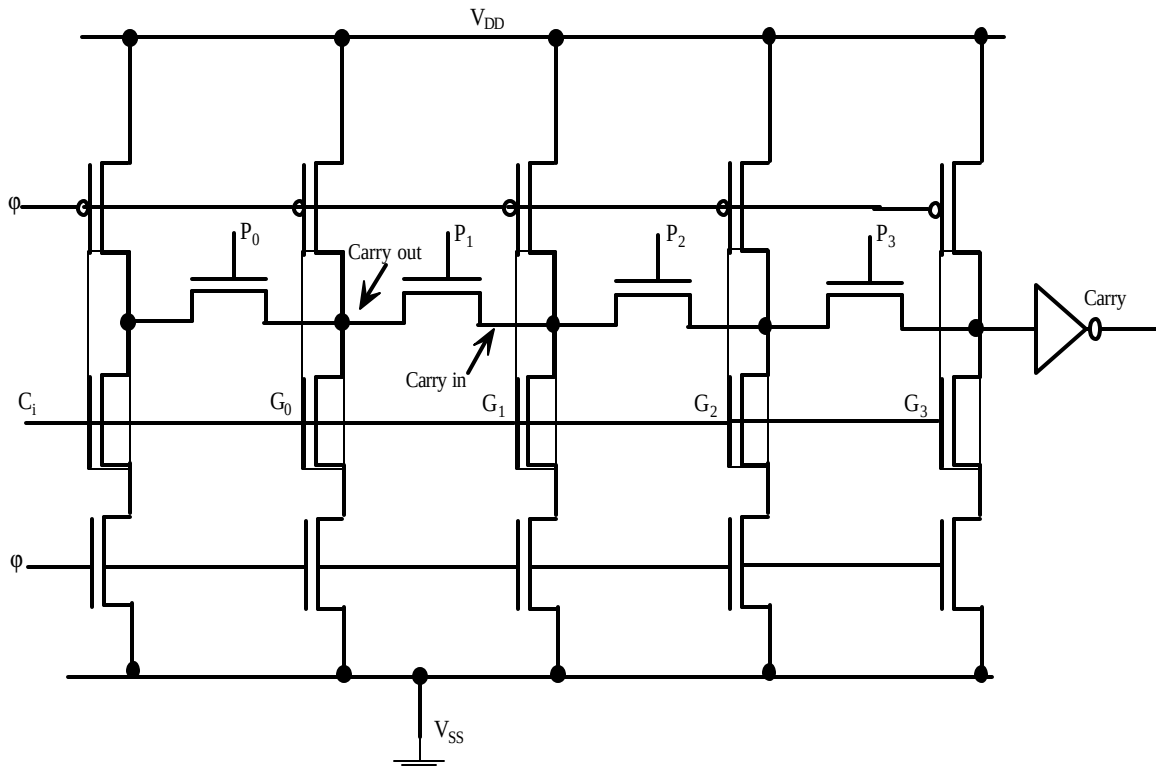


Figure 1.4: Manchester carry chain for a 4-bit adder.

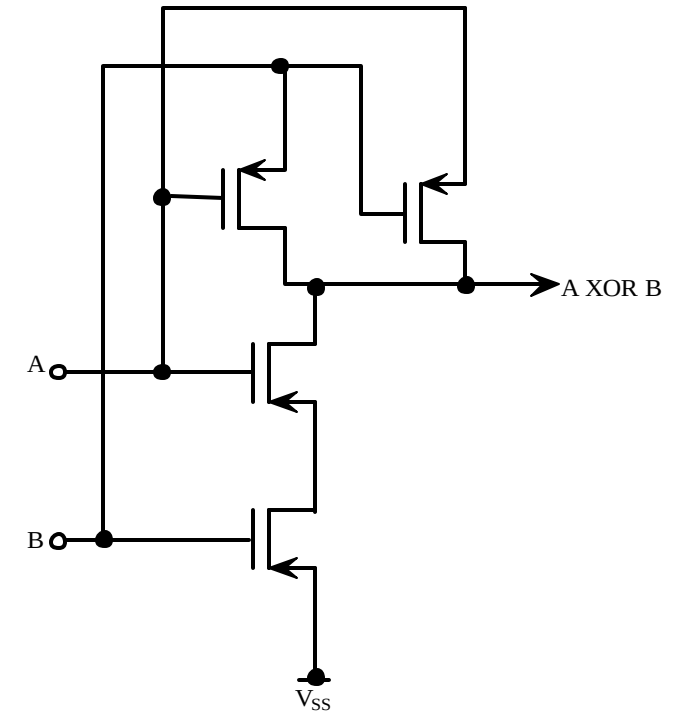
Table 1.1 Number of additional gates and transistors required for different 4-bit full adders [8]

Adder type	**Additional logic required**
Ripple carry	No additional logic required
Carry skip	10 gates + 6 transistors
Carry lookahead	16 gates + 18 transistors
Manchester chain	1 gate + 23 transistors

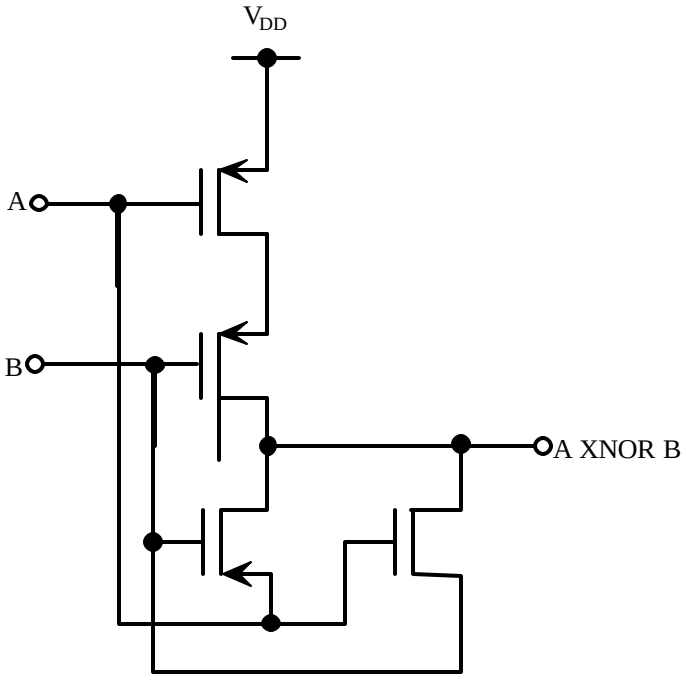
1.1 Literature Review

In the past, ALU and full adder circuits have been implemented for optimum area and delay, each with their distinct features that bring about optimum area and delay. Some of them have been briefly described below to give us an idea of earlier work and shed light on different optimization techniques. Past work related to multiple-input floating gate CMOS applications has been reviewed to give us an idea about its operation, design and simulation issues.

Bui *et al* [9] have designed a low power 10-transistor full adder called Static Energy-Recovery Full-Adder (SERF) using 10 transistors. A novel set of XOR and XNOR gates in combination with existing ones have been used. The XOR and XNOR circuits designed by them do not directly connect to power and ground lines, respectively. It uses four transistors for XOR and XNOR gates in CMOS as shown in Fig. 1.5. Wang *et al* [10] have shown an improved version of XOR and XNOR gates that make use of six transistors as shown in Fig. 1.6. In another design of CMOS 1-bit full adder cell, four transistor XOR and XNOR gates have been used [11]. The cell offers higher speed and lesser power consumption than standard 1-bit full adder cell. Radhakrishnan [12] has presented the design of low power CMOS full adder circuits using transmission function theory. This design uses six transistor CMOS XOR and XNOR gates as shown in Fig. 1.7. Figure 1.8 shows a 16-bit, 2.4ns, 0.5µm CMOS ALU design [13], which consists of a logical and arithmetic unit (LAU), a magnitude comparator (CMP), an overflow detector (OVF) and zero flag detector (ZERO). The ALU employs a binary look-ahead carry (BLC) adder. All units in this design operate in parallel and high speed is achieved.



(a)



(b)

Figure 1.5: Low power CMOS XOR and XNOR gates implemented with 4 transistors [9].

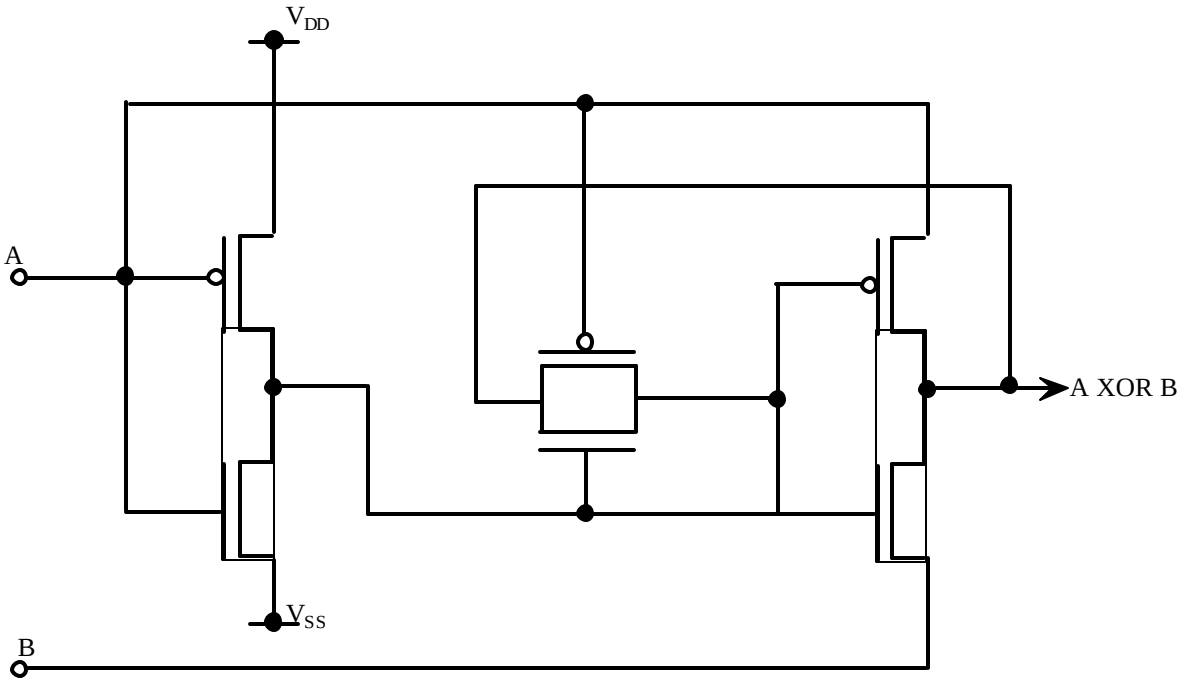


Figure 1.6: A full adder design using six transistors [10].

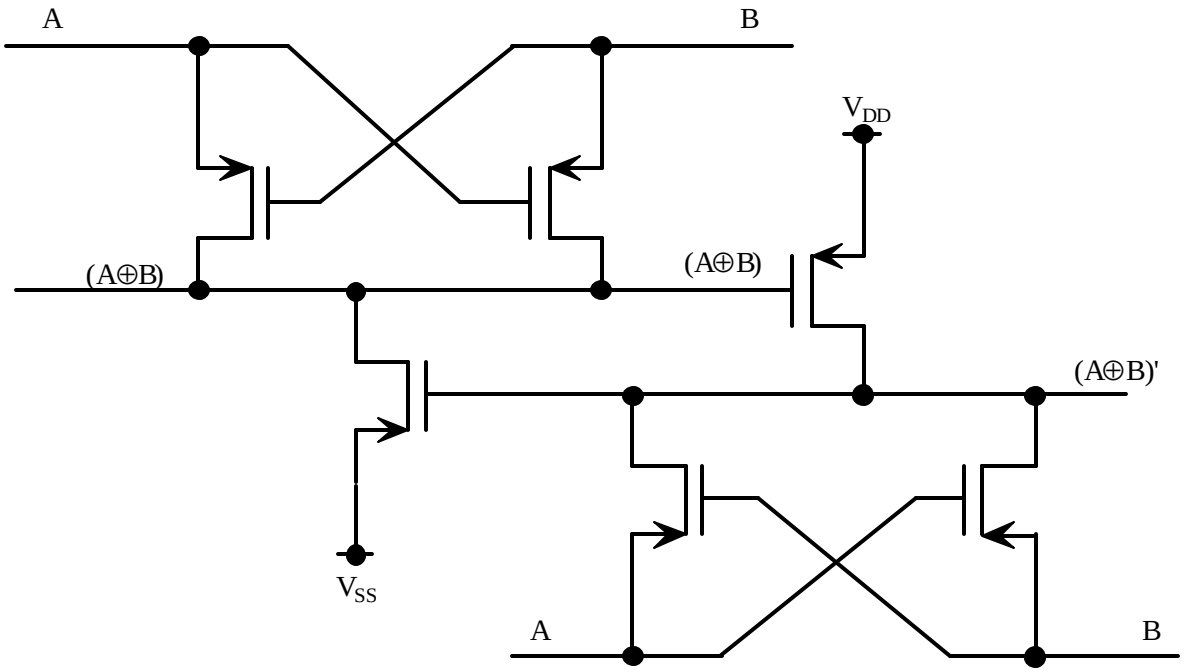


Figure 1.7: Six-transistor CMOS XOR-XNOR gates [12].

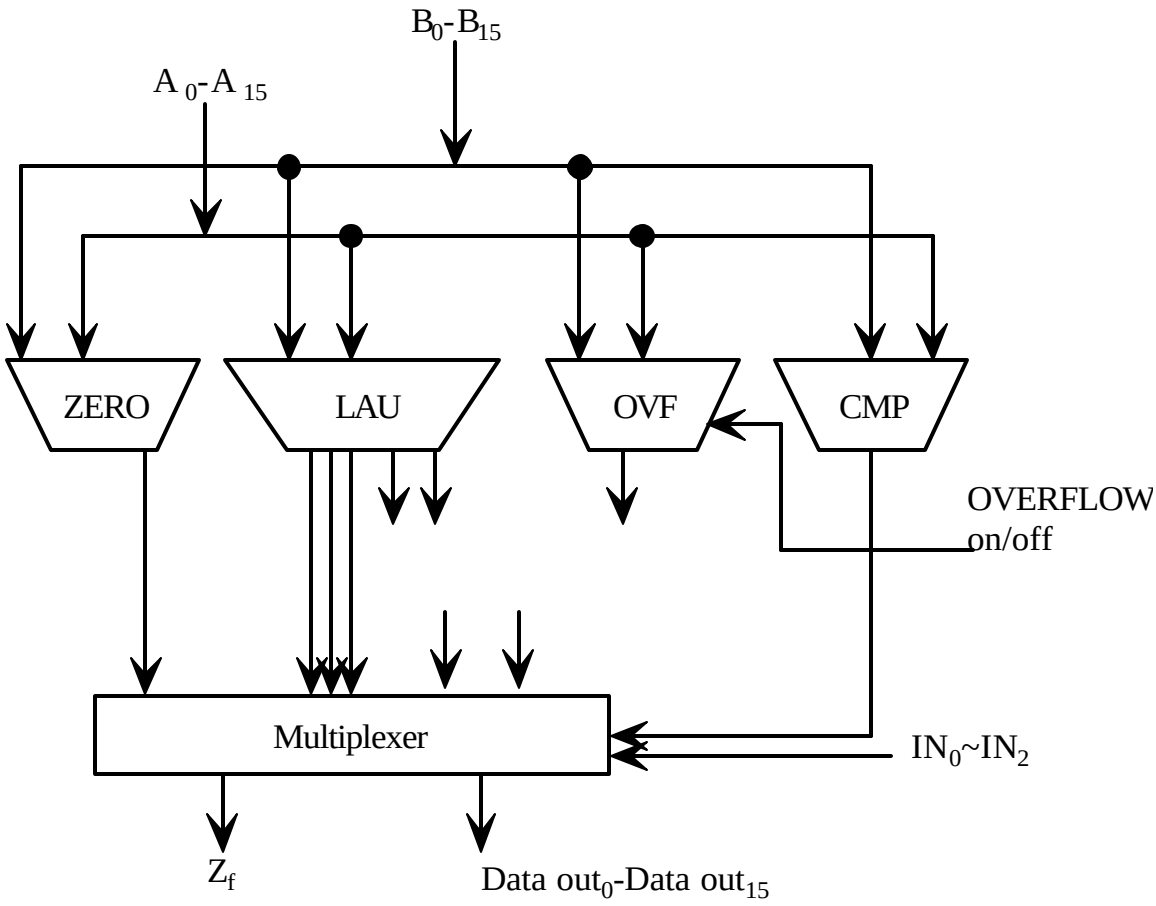


Figure 1.8: 16-bit, 2.4ns, 0.5µm CMOS arithmetic logic unit [13].

In the following, we present review of earlier work in the area of multiple-input floating gate CMOS devices on which our design of a 4-bit ALU is based [14].

Harrison *et al* [15] propose an analog floating memory element for on-chip storage of bias voltages. A floating gate technology has been used to eliminate off-chip biasing voltages in the existing systems by providing these voltages on-chip, with arrays of programmable floating-gate voltages. These arrays can be individually programmed by digital controls. In [16], short-wave ultraviolet light has been applied to floating-gate devices to adjust threshold voltages for optimal performance in a circuit. In [17], a tunable current mirror has been designed where high precision is achieved even with small devices. Floating-gate MOS devices have been used to compute a wide range of translinear functions [18]. In [19], linear amplifiers with rail-to-rail operation with supply voltages less than 1V have been designed in floating gate CMOS.

Floating gate MOS circuits have the inherent ability to adapt to the incoming and outgoing signals by continuously enabling various programming mechanisms. Based on this property, single floating gate FETs, that emulate computational and adaptive properties of biological synoptic elements, were developed [20]. An example of continuously adapting floating-gate circuits is presented in [21]. The autozeroing floating gate amplifier (AFGA) uses tunneling and pFET hot electron injection to adaptively set its dc operating point. No additional circuitry is required.

The application of multi-input floating gate transistor, in designing the full adder, has enabled us to realize the full adder with fewer transistors, when compared to previous designs. In the following chapters, we present the design of a 4-bit ALU for operation at 3.0V, using multiple-input floating gate MOSFETs in standard 1.5µm CMOS process.

1.2 Chapter Organization

Chapter 2 explains the structure and operation of multi-input floating gate MOSFETs and inverter. The design of ALU, its operations and implementation is discussed in chapter 3. This chapter also covers the full adder circuit design implemented using multi-input floating gate technology. The design methodology, technology, post-layout simulations and results are presented in the fourth chapter. Chapter 5 concludes the work. Appendixes A and B summarize the SPICE MOSFET model parameters. In Appendix C, floating gate MOSFET simulation in SPICE is presented.

Chapter 2

Multi-Input Floating Gate MOSFET (MIFG MOSFET)

2.1 Introduction

Multi-input floating gate MOSFET is basically an enhancement of the basic MOSFET. Multi input floating gate MOSFET is defined as a transistor that switches to an ON or OFF state depending on the weighted sum of all input signals applied at its input gate [22]. A number of input voltage signals are capacitively coupled to the input gate. The voltage on the gate is the result of the weighted sum of the input voltages. The output of the transistor is pulled to a 'HIGH' or 'LOW' state depending on the potential of the floating gate being lesser or greater than the threshold voltage of the device. This weighted sum calculation method of realizing logic functions is also termed as multiple value logic [23]

There are certain advantages in using multiple-valued logic for VLSI applications. Any given function realized using multi-input floating gate transistors uses fewer transistors and interconnections resulting in small area consumption. The multi-input floating gate MOSFET operation depends on the weighted sum of voltages at input nodes, which are capacitively coupled to the gate. This leads to only a very negligible amount of charging and discharging currents and results in low power dissipation. Multi-input floating gate MOSFETs have been used to great advantage in digital and analog VLSI applications including analog filters, analog multipliers, data converters, EPROMs, and EEPROMs [23-29].

2.2 MIFG MOSFET Structure and Principle

The structure of multi-input floating gate MOSFET comprises of the floating gate and number of input gates built on poly2, which is coupled to poly1 gate by capacitors between poly1 and poly2. This structure makes it possible for multi-input floating gate devices to be implemented in double polysilicon CMOS process. The basic structure of a multi-input floating gate MOSFET is presented in Fig. 2.1 [3]. The floating gate in the MOSFET extends over the channel and the field oxide. A number of control gates, which are inputs to the transistor, are formed over the floating gate using a second polysilicon layer (poly 2). Figure 2.2 shows capacitive coupling between the multiple-input gates and floating gate and the channel. In Fig. 2.2, C_1, C_2, C_3, …, C_n, are the coupling capacitors between the floating gate and the inputs. The corresponding terminal voltages are V_1, V_2, V_3, …, V_n, respectively. C_0 is the capacitor between the floating gate and substrate. V_{SS} is the substrate voltage. Q_1, Q_2, Q_3, …, Q_n are the charges stored in corresponding capacitors C_1, C_2, C_3, …, C_n. At any instant, net charge $Q_F(t)$ on the floating gate is given by the following equations [3, 30]:

$$Q_F(t) = Q_0 + \sum_{i=1}^{n}(-Q_i(t)) = \sum_{i=0}^{n} C_i(\Phi_F(t) - V_i(t)) \tag{2.1}$$

$$\text{or } Q_F(t) = \Phi_F(t)\sum_{i=0}^{n} C_i - \sum_{i=0}^{n} C_i V_i(t) \tag{2.2}$$

Where n is the number of inputs, Q_0 is the initial charge present on the floating gate, $Q_i(t)$ is the charge present in capacitor C_i and $\Phi_F(t)$ is the potential at the floating gate.

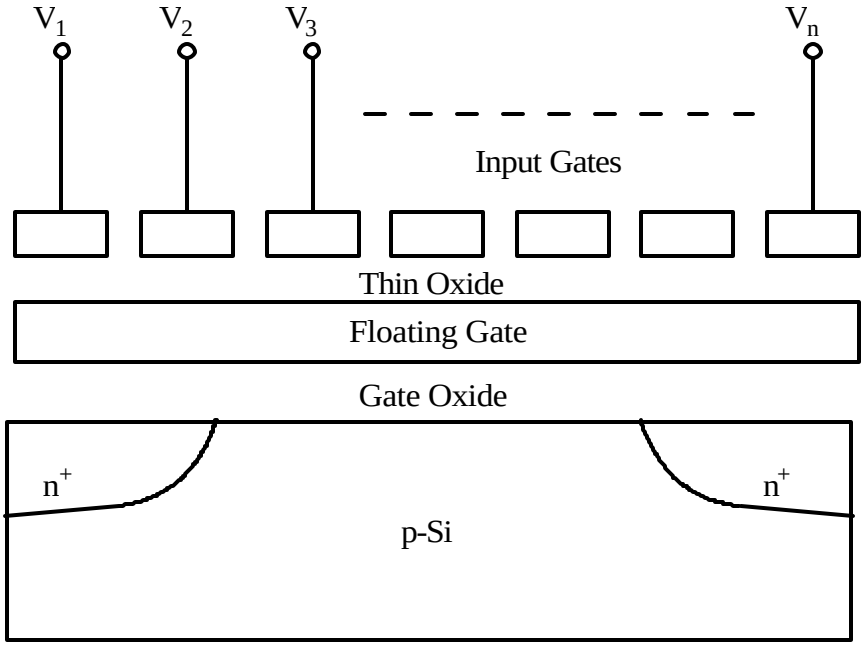


Figure 2.1: Basic structure of a multi-input floating gate MOSFET.

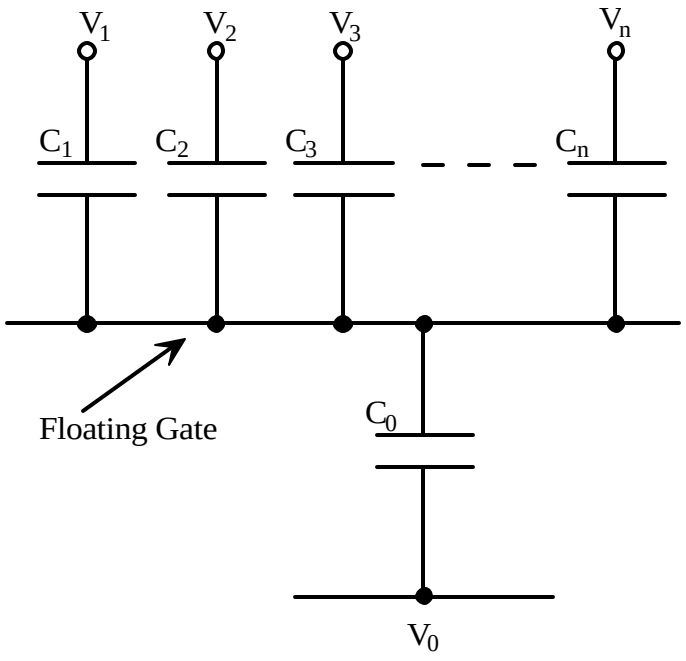


Figure 2.2: Terminal voltages and coupling capacitances of a multi-input floating gate MOSFET.

Setting V_{SS} = 0V and applying the law of conservation of charge at the floating gate, Eq. (2.2) can be expressed as follows:

$$\Phi_F(0)\sum_{i=0}^{n} C_i - \sum_{i=1}^{n} C_i V_i(0) = \Phi_F(t)\sum_{i=0}^{n} C_i - \sum_{i=1}^{n} C_i V_i(t), \tag{2.3}$$

Or

$$\Phi_F(t)\sum_{i=0}^{n} C_i - \Phi_F(0)\sum_{i=0}^{n} C_i = \sum_{i=1}^{n} C_i V_i(t) - \sum_{i=1}^{n} C_i V_i(0), \tag{2.4}$$

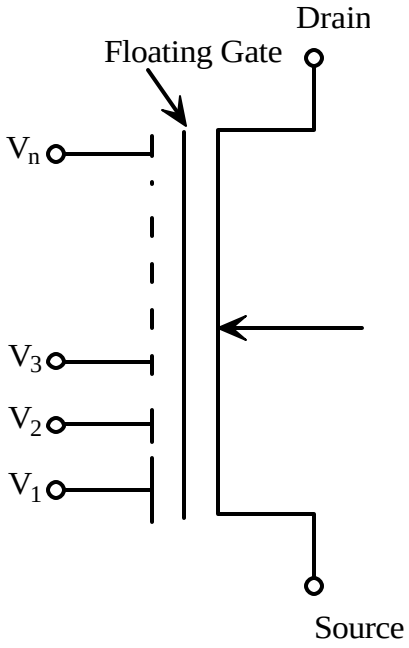
Or

$$\Phi_F(t) = \Phi_F(0) + \frac{\sum_{i=1}^{n} C_i V_i(t) - \sum_{i=1}^{n} C_i V_i(0)}{\sum_{i=0}^{n} C_i} \tag{2.5}$$

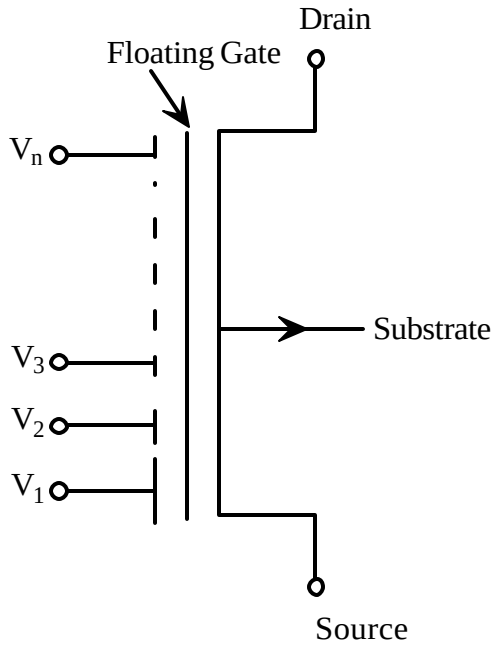
Assuming zero initial charge on the floating gate in Eq. (2.2), Eq. (2.5) reduces to

$$\Phi_F(t) = \frac{\sum_{i=1}^{n} C_i V_i(t)}{\sum_{i=0}^{n} C_i} \tag{2.6}$$

Switching ON or OFF of the n-MOSFET depends on whether $\Phi_F(t)$ is greater than or less than the threshold voltage of the transistor. The value of Φ_F determined by Eq. (2.6) holds true, as long as all the input capacitive coupling co-efficients remain unchanged during device operation. The oxide capacitance C_0 is assumed to remain constant. Figure 2.3(a-b) shows the symbols of multi-input floating gate n- and p-MOSFETs.



(a)



(b)

Figure 2.3: (a) Multi-input floating gate n-MOSFET. (b) Multi-input floating gate p-MOSFET.

Let us denote V_{th} as the threshold voltage of the transistor. Then the transistor turns on at the condition $\Phi_F > V_{th,}$ and is described by the following equation.

$$\frac{V_1 \times C_1 + V_2 \times C_2 + V_3 \times C_3 + ...V_n \times C_n}{C_1 + C_2 + C_3 + ...C_n + C_o} > V_{th} \qquad (2.7)$$

2.3 Multiple-Input Floating Gate CMOS Inverter

Multiple-input floating gate CMOS inverter is shown in Fig. 2.4 [3]. V_1, V_2, V_3,…,V_n are input voltages and C_1, C_2, C_3 ,…,C_n are corresponding input capacitors. Eq. (2.6) is used to determine voltage on the floating gate of the inverter. Weighted sum of all inputs is performed at the gate and is converted into a multiple-valued input voltage, V_{in} at the floating gate. The switching of the floating gate CMOS inverter depends on whether V_{in} obtained from the weighted sum, is greater than or less than the inverter threshold voltage or inverter switching voltage (Φ_{in}). The switching voltage is computed from the voltage transfer characteristics of a standard CMOS inverter and is given by the following equation [5].

$$\Phi_{inv} = \frac{(\Phi_{go} + \Phi_{s1})}{2} \qquad (2.8)$$

Where Φ_{g0} and Φ_{s1} are the input voltages at which V_{out} is V_{DD}-0.1V and 0.1V, respectively. Hence, the output (V_{out}) of a multi-input floating gate CMOS inverter is

V_{out} = HIGH (3V) if $\Phi_F < \Phi_{inv}$

= LOW (0V) if $\Phi_F > \Phi_{inv}$ (2.9)

The capacitance network in an n-input floating gate CMOS inverter is shown in Fig. 2.5 [31].

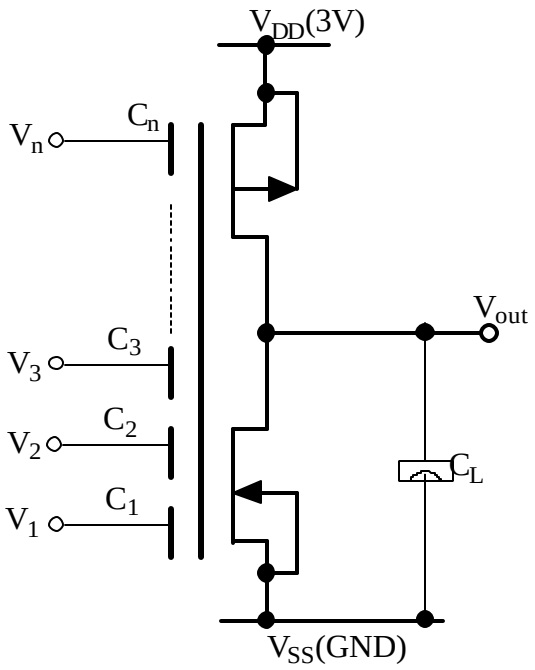


Figure 2.4: Multi-input floating gate CMOS inverter.

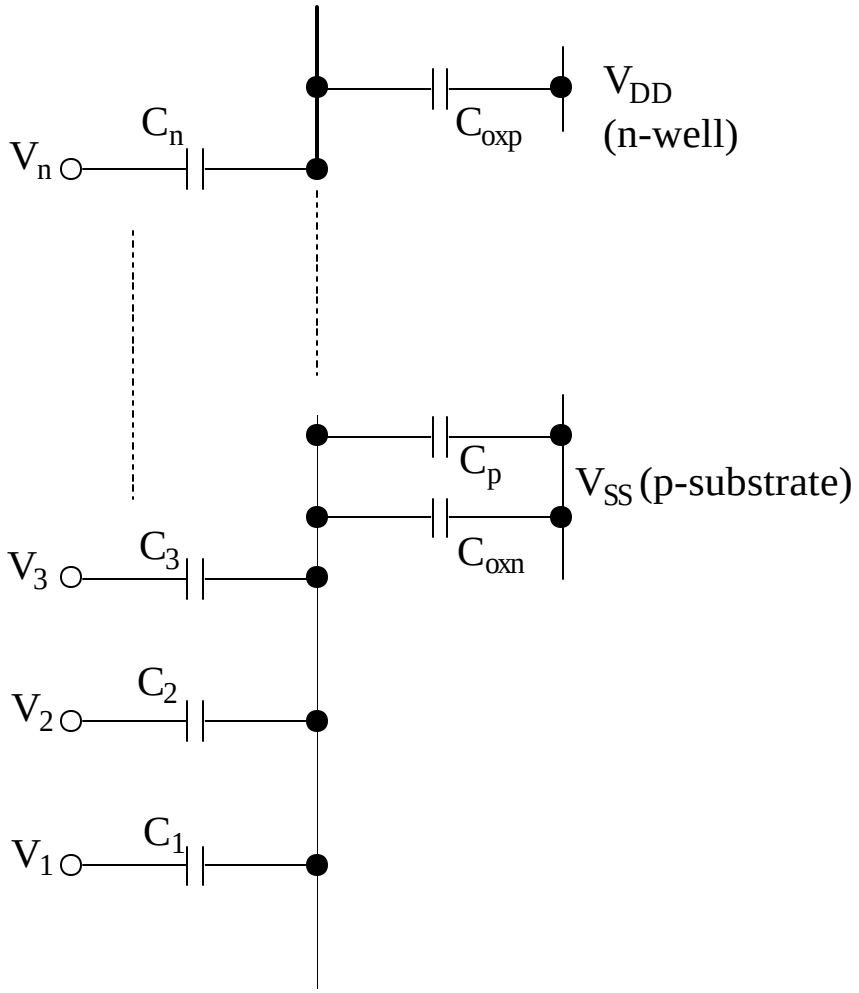


Figure 2.5: Capacitive networks formed for a multi-input floating gate CMOS inverter.

The gate oxide capacitance of a p-MOSFET, C_{oxp} is between the floating gate and n-well and is connected to V_{DD}. C_{oxn} is between the floating gate and p-substrate and is connected to V_{SS}. C_p is the capacitance formed between polysilicon floating gate and substrate. At this point, we should note the substrate potential so far has been assumed to be zero. But this is not so when it comes to the p-channel MOSFET in the inverter because the n-well is biased to V_{DD} . Equation (2.6) is modified as follows.

$$\Phi_F(t) = \frac{\sum_{i=1}^{n} C_i V_i(t) + C_{0xp} V_{DD}}{\sum_{i=0}^{n} C_i} \tag{2.10}$$

Voltage on the floating gate is given by:

$$\Phi_F = \frac{V_1 \times C_1 + V_2 \times C_2 + V_3 \times C_3 + \ldots V_n \times C_n + V_{DD} \times C_{oxp} + V_{SS} \times (C_p + C_{oxn})}{C_1 + C_2 + C_3 + \ldots C_n + C_{oxn} + C_{oxp} + C_p} \tag{2.11}$$

Here V_{SS} is taken to be the reference ground potential. Equation (2.11) becomes,

$$\Phi_F = \frac{V_1 \times C_1 + V_2 \times C_2 + V_3 \times C_3 + \ldots V_n \times C_n + V_{DD} \times C_{oxp}}{C_1 + C_2 + C_3 + \ldots C_n + C_{oxn} + C_{oxp} + C_p} \tag{2.12}$$

In Fig. 2.6, a 4-input CMOS inverter is shown for which a layout was drawn and simulation was done on the extracted SPICE netlist. All four input capacitors' value was chosen to be 100 fF each. The input voltages to the capacitors are pulse waveforms with varying pulse widths and time periods. Figure 2.7 shows the layout of a 4-input floating gate CMOS inverter. A transient analysis is performed for the inverter for four different inputs. The maximum pulse voltage of the input waveforms is 3.0V. The switching threshold voltage of the floating gate MOSFET can be estimated from Eq. (2.13) as given below [32].

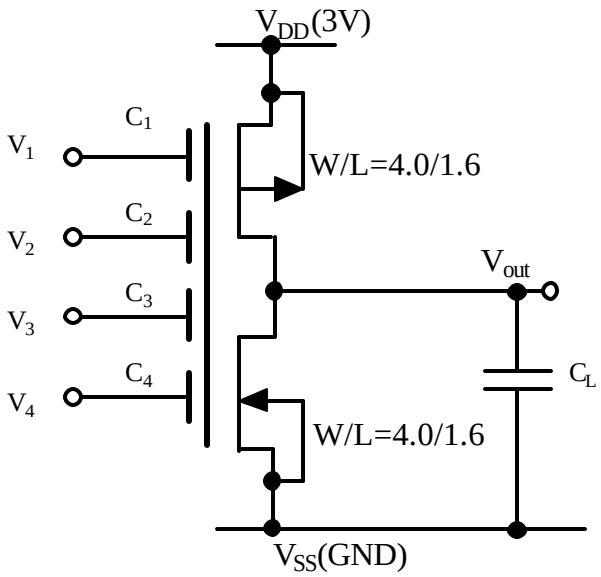


Figure 2.6: A 4-input floating gate CMOS inverter.

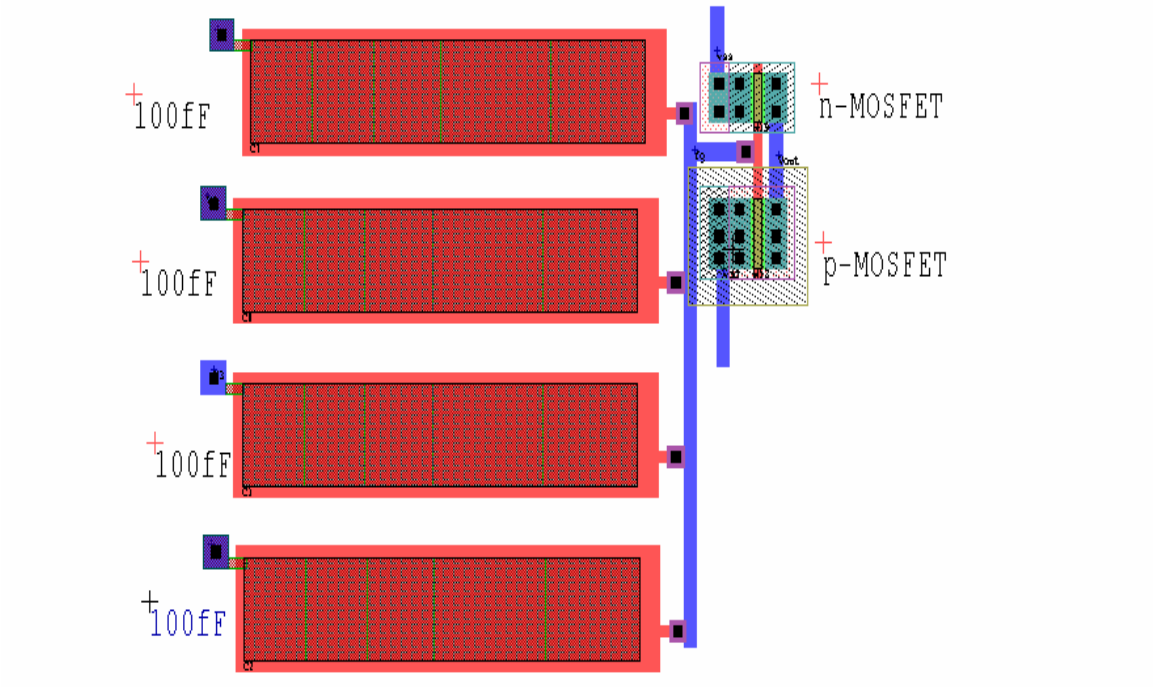


Figure 2.7: Layout of a 4-input floating gate CMOS inverter
Note: The capacitor dimension is 82.8µm x 18.4µm.

$$V_{inv} = \frac{V_{DD} + \sqrt{W_n \mu_n L_n / W_p \mu_p L_p} V_{Tn} + V_{Tp}}{\sqrt{W_n \mu_n L_n / W_p \mu_p L_p} + 1} \quad (2.13)$$

The values of V_{Tn} and V_{Tp} are obtained from SPICE parameters listed in Appendix A. The mobility ratio for n-MOSFET and p-MOSFET is assumed to be 2. The lengths L_n and L_p are set to minimum value used in 1.5µm n-well CMOS process. The values for W_p and W_n are calculated as 3.2µm and 10.4µm, respectively to satisfy $V_{inv} = V_{DD}/2$. Figure 2.8 shows the simulated transfer characteristics of a 4-input floating gate CMOS inverter [32]. As the input voltage changes sequentially the output voltage varies. This corresponds to the steps seen in the output waveform of Fig. 2.8. Considering that we use this inverter for a digital application where the outputs need to be either '1' or '0', we see that the output is a perfect '1' only when all 4 inputs are '0'(0V). It follows that the output is a perfect '0' when all 4 inputs are '1'(3.0V).

2.4 Variable Threshold Voltage

The uniqueness of multi-input floating gate inverter lies in the fact that the switching voltage can be varied by selection of those capacitor values through which the inputs are coupled to the gate. Ordinarily, varying the W_p/W_n ratios of the inverter does the adjustment of threshold voltage. In multi-input floating gate inverter, varying the coupling capacitances to the gate can vary the switching point in DC transfer characteristics.

Figure 2.9(a) shows a 3-input floating gate CMOS inverter. The input capacitors are C_1, C_2 and C_{in}. Capacitors C_1 and C_2 are connected permanently to V_{DD} and V_{SS}, respectively. A 3.0V DC voltage is applied to input capacitor C_{in}. In this experiment, the values of C_1, C_2 and C_{in} were initially fixed at 100 fF each. The bias voltages, V_{DD} and V_{SS} are 3.0 V and 0 V, respectively.

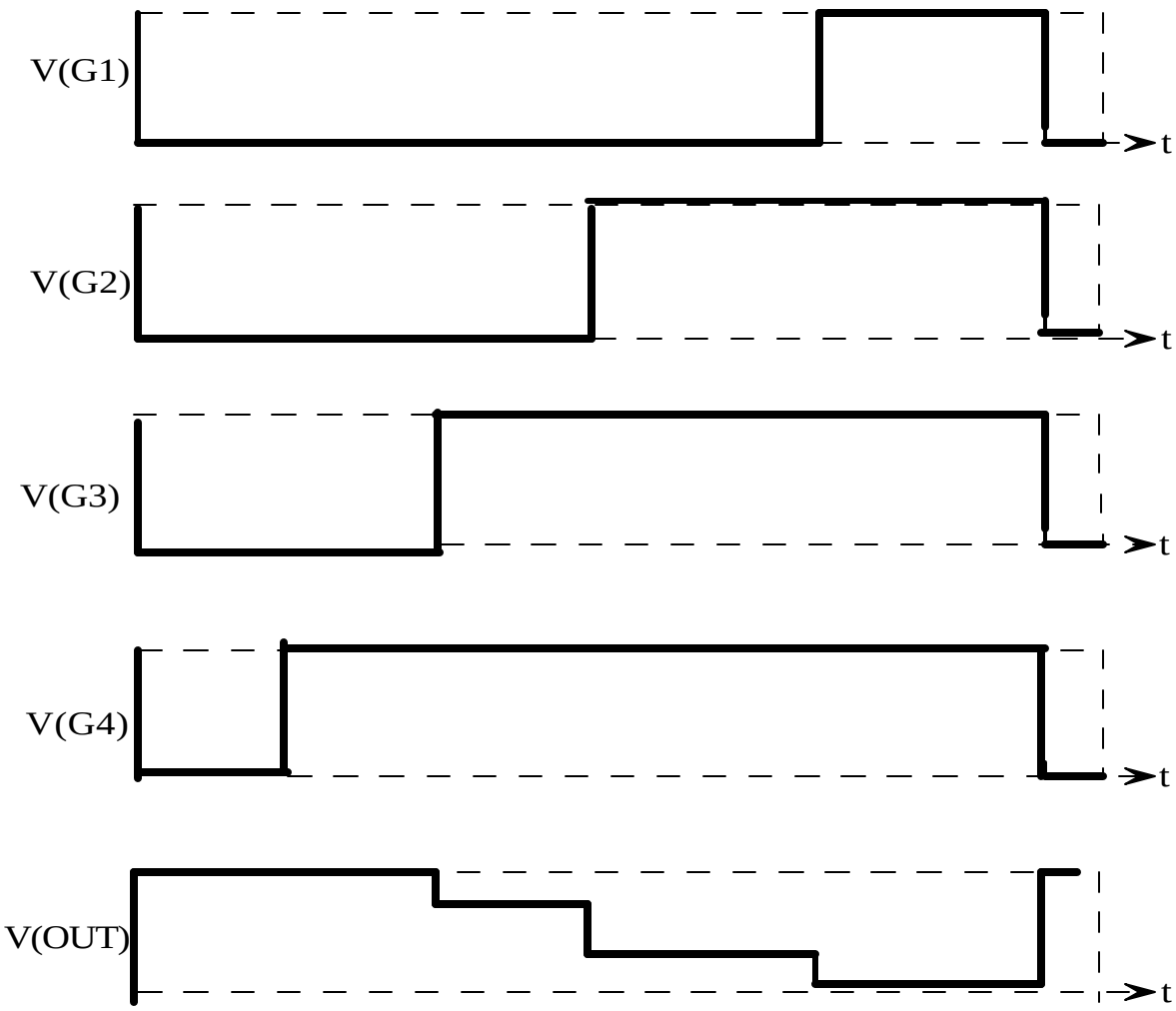
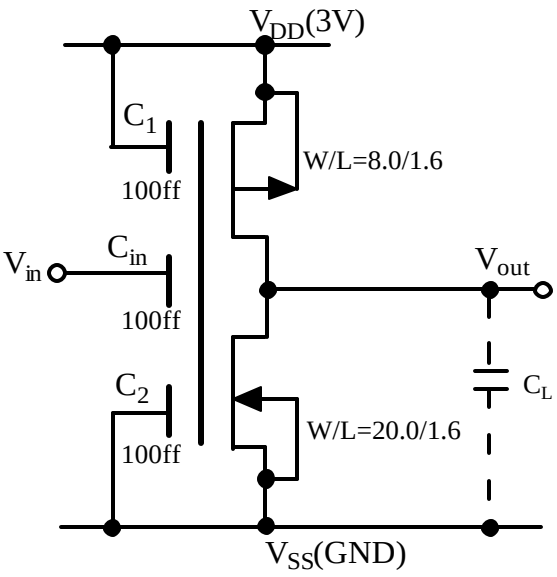


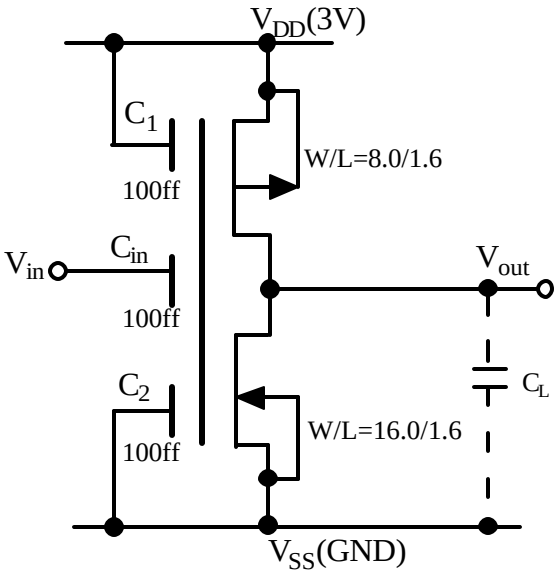
Figure 2.8: Transfer characteristics of a 4-input floating gate CMOS inverter.

The W_p/W_n ratio of the MOSFETs of the floating gate CMOS inverter is 8.0/20. Figures 2.9(b-d) are essentially similar to Fig. 2.9(a) but the W_p/W_n ratios of the inverters are varied as shown in the figure. The input voltage given to each of the four circuits is the same. A DC analysis was performed using SPICE for all four configurations of Fig. 2.9. Figure 2.10 shows the voltage transfer characteristics for all four configurations of Fig. 2.9. The switching voltage on each of the voltage transfer characteristics is marked. From the voltage transfer characteristics of Fig. 2.10, we see that as W_p/W_n ratio of the inverter increases, the switching voltage moves to the left. This can be explained by the shift in transfer characteristics to right as W_p/W_n ratio decreases [33].

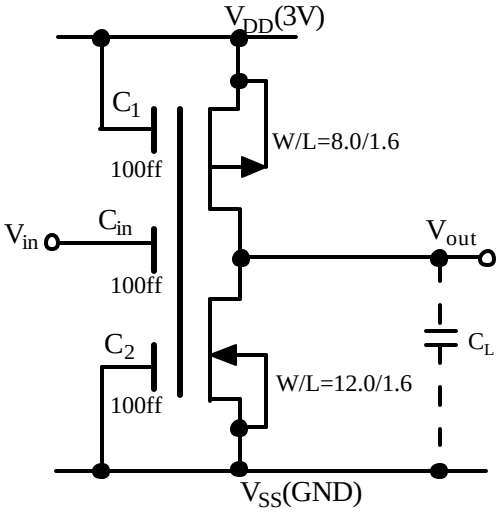
We repeat the simulation for another set of inverters shown in Fig. 2.11, where the W_p/W_n ratio is maintained constant at 8.0/16.0. However, the size of capacitor C_1 is increased by 100fF progressively. The same input voltage as in Fig. 2.9 is applied to the circuits. Figure 2.12 shows the DC voltage transfer characteristics for all four configurations of Fig. 2.11. From the voltage transfer characteristics, we notice that as the capacitor ratio C_1/C_2 increases, the output waveform shifts in to the right. This can be explained from Eq. (2.12) for multi-input floating gate CMOS inverter. The change in coupling capacitor ratio C_1/C_2 changes the potential ϕ_F on the floating gate. The output switches its state at different input voltages just as it would if the W_p/W_n ratio were to be varied. The switching voltages marked on each of the voltage transfer characteristics are calculated from Eq. (2.8).



(a)



(b)



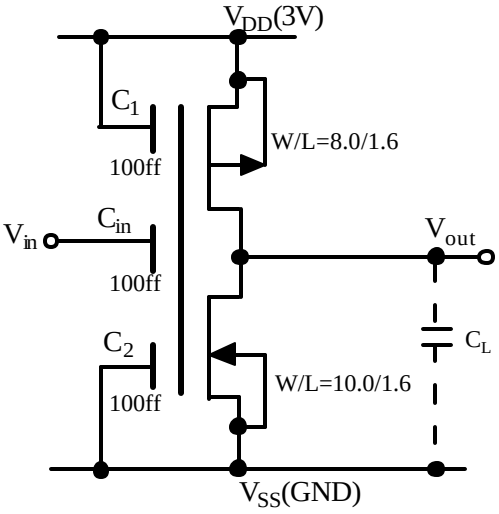


Figure 2.9(a-d): 3-input CMOS inverter with W_p/W_n ratios of 0.4, 0.5, 0.67 and 0.8 , respectively.

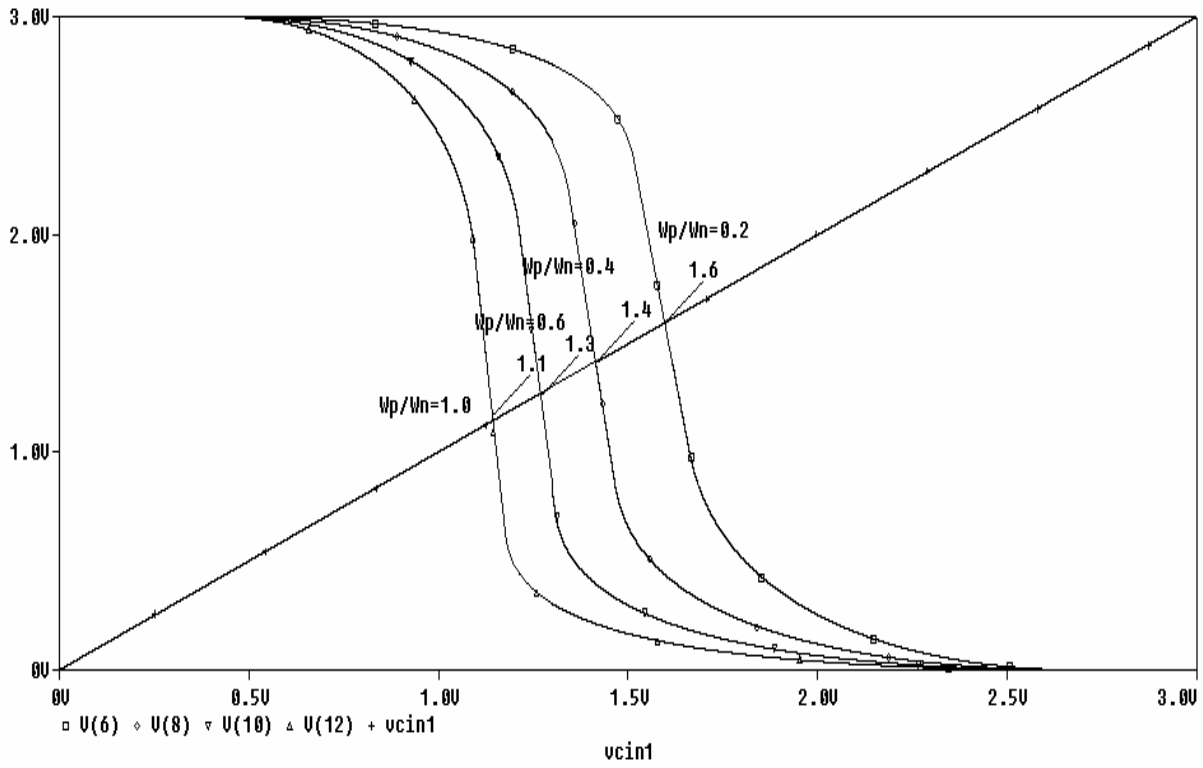


Figure 2.10: Transfer characteristics of 3-input floating gate CMOS inverters shown in Fig. 2.9 with W_p/W_n ratios of 0.4, 0.5, 0.67 and 0.8.

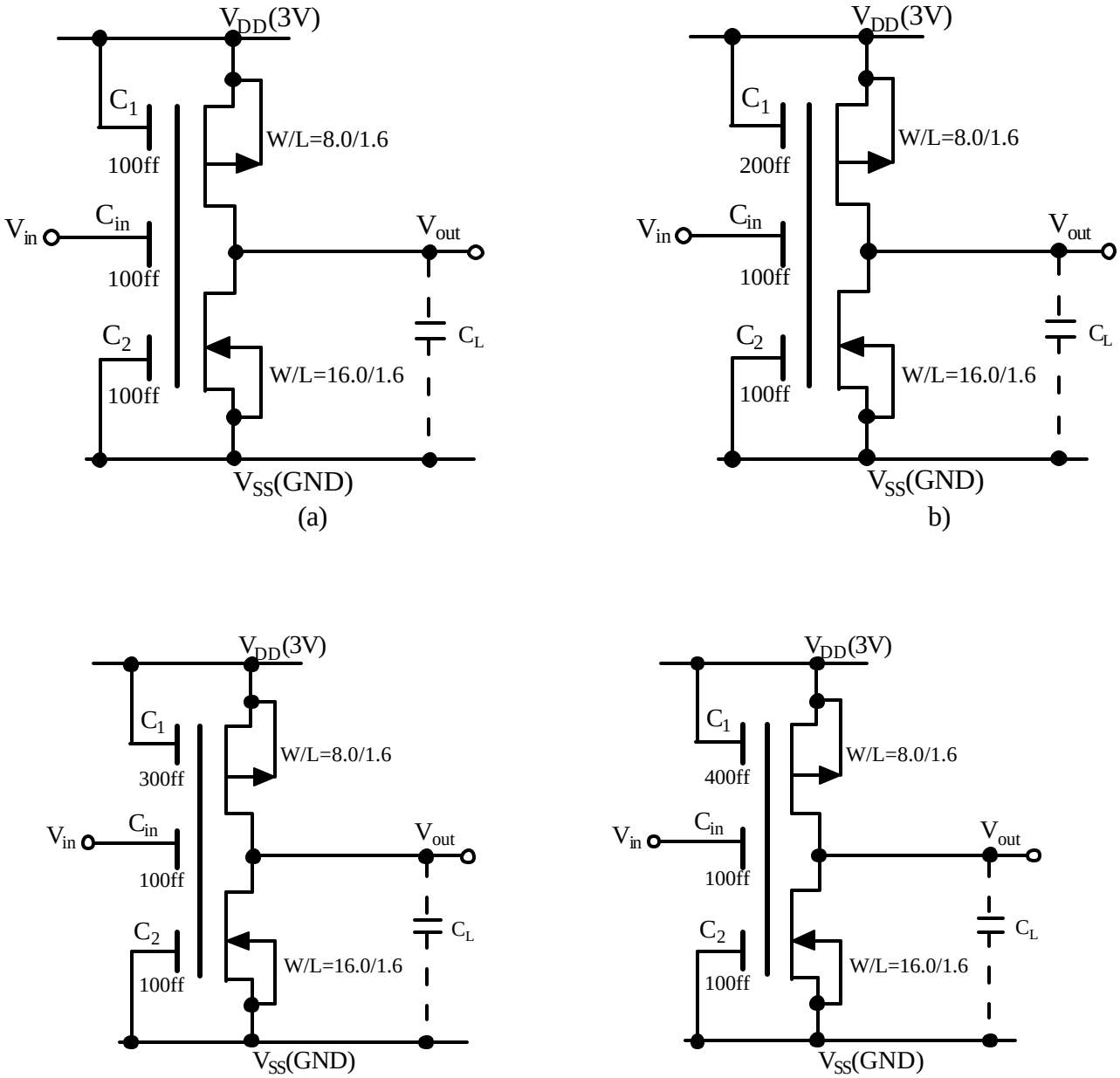


Figure 2.11(a-d): 3-input floating gate CMOS inverter with capacitor C_1 varied from 100fF to 400fF.

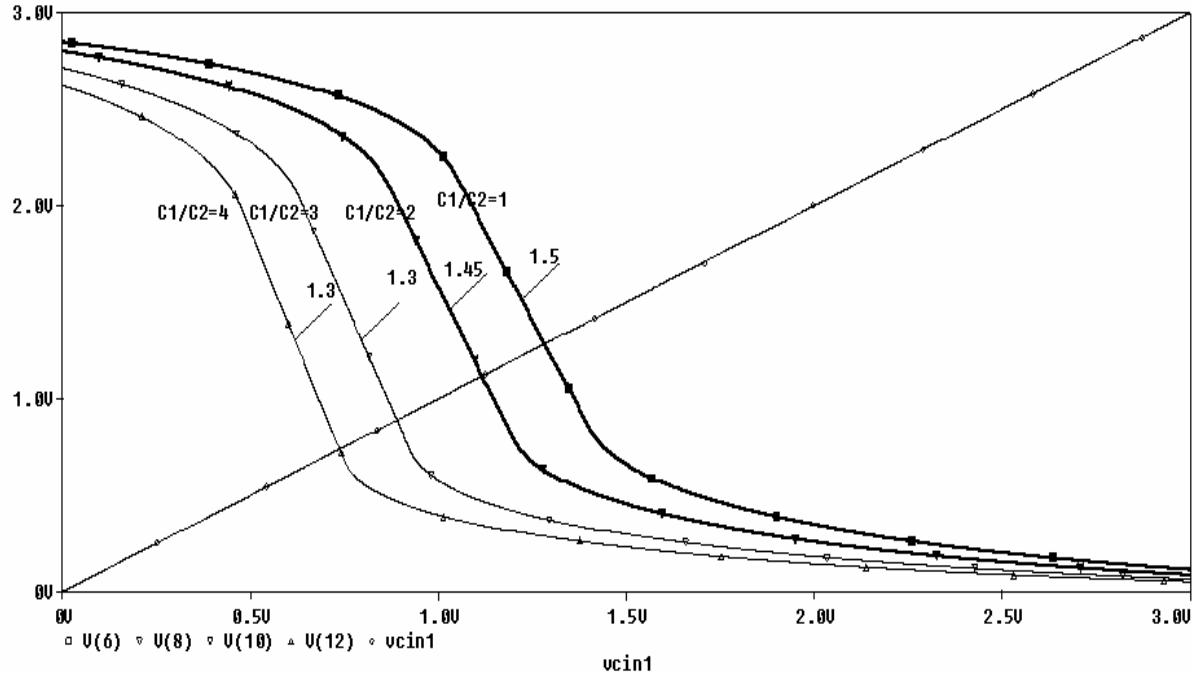


Figure 2.12: Output waveforms for CMOS inverters with capacitor ratio C_1/C_2 varied from 1 to 4.

2.5 Unit Capacitance

The floating gate CMOS circuit design layout faces certain shortcomings in fabrication process. Due to fabrication process variations in runs employed, designed capacitors may not turn out to be of right values. Some of the floating gate designs are sensitive to capacitor changes and this may change the output values. The key factor in designing capacitor values for multi-input floating gate designs is to start with a unit capacitance value [34]. The rest of the capacitors in the circuits are either unit value or integral multiples of it. The area capacitance parameter between poly1 and poly2 layers in 1.5μm standard CMOS process [35] varies from 580aF/μm^2 to 620aF/μm^2 for different runs. Since we do not have prior knowledge of which run would be used in fabrication of our chip an average value has been used. In our design we have used 596aF/μm^2.

Table 2.1 shows the minimum and maximum capacitance values obtained considering variations in area capacitance parameter. This table also includes the capacitor values in case of edges being shortened by 0.5μm and 1.0μm. The percentage variation in capacitance from desired value and the worst-case scenario of edges being reduced by 1μm is calculated.

For 10fF $\quad \%\text{change} = \dfrac{9.536 - 2.32}{9.536} \times 100 = 76\%$

For 20fF $\quad \%\text{change} = \dfrac{21.456 - 9.28}{21.456} \times 100 = 57\%$

For 100fF $\quad \%\text{change} = \dfrac{100.724 - 70.18}{100.724} \times 100 = 30\%$

For 500fF $\quad \%\text{change} = \dfrac{501.236 - 364.5}{501.236} \times 100 = 27\%$

.

Table 2.1.Variation in unit capacitance with respect to area and capacitance
Note: C' is the capacitance per unit area

Unit Capacitance	Area Required	C' 580 aF/μm^2	C' 596 aF/μm^2	C' 620 aF/μm^2
10 fF	4 μ x 4 μ	9.280fF	9.536 fF	9.920 fF
	3 μ x 3 μ	5.220fF	5.364 fF	5.580 fF
	2 μ x 2 μ	2.320fF	2.384 fF	2.480 fF
20fF	6 μ x 6 μ	20.880fF	21.456 fF	22.320 fF
	5 μ x 5 μ	14.5fF	14.9 fF	15.5 fF
	3 μ x 3 μ	9.28fF	9.536 fF	9.920 fF
100fF	13 μ x 13 μ	98.020fF	100.724 fF	104.780 fF
	12 μ x 12 μ	83.520fF	85.824 fF	89.280 fF
	11 μ x 11 μ	70.180fF	72.116 fF	75.020 fF
500fF	29 μ x 29 μ	487.78fF	501.236 fF	521.420 fF
	28 μ x 28 μ	454.72fF	467.264 fF	486.080 fF
	27 μ x 27 μ	364.5fF	434.484 fF	451.980 fF

From the above calculation, it is seen that, greater the unit capacitance value when compared to gate capacitance value, lesser is the variation. The probability of the functionality being affected reduces significantly in this case. To maintain the percentage change in capacitor value, due to fabrication process, within 30% we need to use a unit capacitance value of atleast 500 fF.

2.6 Design Issues

An important factor affecting the accuracy of inversion threshold voltage is the presence of residual charge on floating gate after fabrication processes. The residual charge alters Q_F, the charge on floating gate that is responsible for fixing the inversion threshold voltage. This may vary from device to device also causing inaccuracies in output logic levels desired. However, such residual charges can be reduced by exposure to ultraviolet light irradiation. With such UV irradiation technique, it has been shown that initially fluctuating voltage converges to one final steady value [18, 28].

Floating gate CMOS inverter gives a degraded output level when inputs are not uniform. Consider a multi-input floating gate CMOS inverter with three inputs with coupling capacitances of 500fF each. When all three input are at logic 'HIGH' (3 V) and all three inputs are at logic 'LOW' (0V) then the voltage on the floating gate, ϕ_F, is calculated using Eq.(2.12), as shown below.

$$\Phi_F = \frac{500\times 3 + 500\times 3 + 500\times 3}{1500 + 30} = 2.9\text{V}$$

where C_{ox} is approximated to 30ff.

$$\Phi_F = \frac{500\times 0 + 500\times 0 + 500\times 0}{1500 + 30} = 0\text{V}$$

When one of the inputs is logic 'HIGH' or two of the inputs are 'HIGH' then the voltage on floating gate is calculated as follows,

$$\Phi_F = \frac{500\times 3 + 500\times 3 + 500\times 0}{1500 + 30} = 1.96\text{V}$$

$$\Phi_F = \frac{500\times 0 + 500\times 0 + 500\times 3}{1500 + 30} = 0.3\text{V}$$

In these cases described above, the output does not swing completely between 3 V and 0V as all inputs do not have the same logic level. Figure 2.13 shows simulated waveforms for the above mentioned inverter. V_{C1}, V_{C2} and V_{C3} are the input waveforms and V_{OUT} is the output waveform. From Fig. 2.13 we see that when all three inputs are at '0' the output voltage swings between 0V and 3V, but when one of the inputs is logic 'HIGH' or two of the inputs is 'HIGH' the output voltage swing is between 0V and 2.5V. In such a case the output needs to be buffered to give complete swing.

Simulation techniques used for multi-input floating gate CMOS circuits are different from a standard CMOS inverter. Simulation using SPICE gives the problem of DC convergence. It views the capacitors as open circuits initially and stops the simulation run. To overcome the problem different approaches have been explained in [6, 7, 36]. These techniques employ additional use of resistors and voltage controlled voltage sources (VCVS) for specifying the initial floating gate voltage. We have used the method suggested by Yin *et al* [6], which is described in Appendix C.

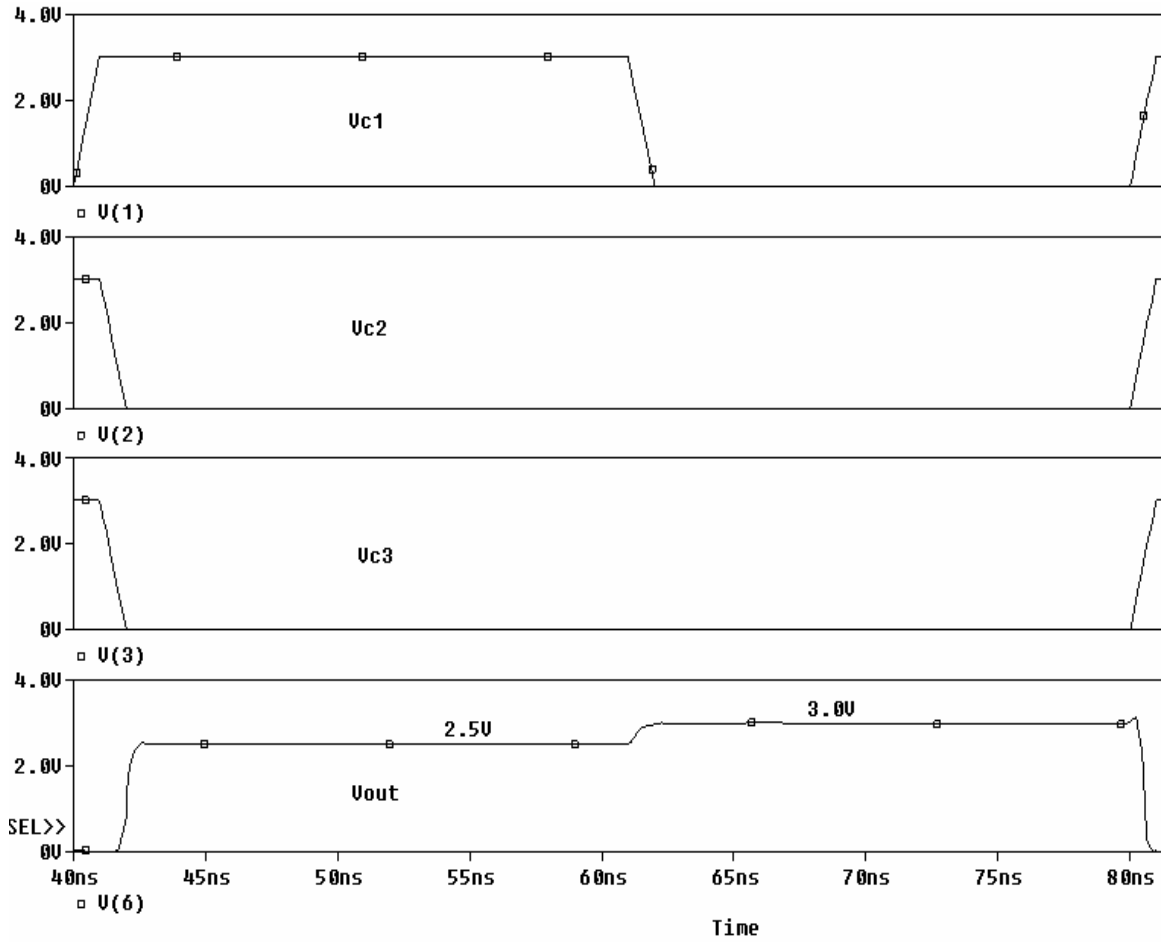


Figure 2.13. Input and output waveforms for a 3-input floating gate CMOS inverter showing degraded output of 2.5V as well as perfect output of 3.0V. Note: W/L ratio of the n-MOSFET = 4.0/1.6, W/L of p-MOSFET = 8.0/1.6, C_L = 0.1pF.

Chapter 3

The ALU Design

3.1 The ALU Design and Operation

A 4-bit ALU has been designed for 3.0 V operation in which, the full adder design has been implemented using MIFG CMOS inverters. The ALU has four stages, each stage consisting of three parts: a) input multiplexers b) full adder and c) output multiplexers. The ALU performs the following four arithmetic operations, ADD, SUBTRACT, INCREMENT and DECREMENT. The four logical operations performed are EXOR, EXNOR, AND and OR. The input and output sections consist of 4 to 1 and 2 to 1 multiplexers. The multiplexers were designed using the pass transistor logic. A set of three select signals has been incorporated in the design to determine the operation being performed and the inputs and outputs being selected. Figure 3.1 shows the 4-bit ALU with the CARRY bit cascading all the way from first stage to fourth stage. In Fig. 3.1, the ALU design consisting of eight 4 to 1 multiplexers, eight 2 to 1 multiplexers and four full adders. The 4-bit ALU was designed in 1.5µm, n-well CMOS technology. This chapter explains in detail the 4-bit ALU design. All of the multiplexers have been implemented using pass transistors, and the full adder alone has been designed using MIFG CMOS logic. Each stage is discussed in detail in the further sections of this chapter.

3.1.1 Multiplexer Design

The multiplexers have been used in the ALU design for input and output signals selection. The multiplexer is implemented using pass transistors [33, 37]. This design is

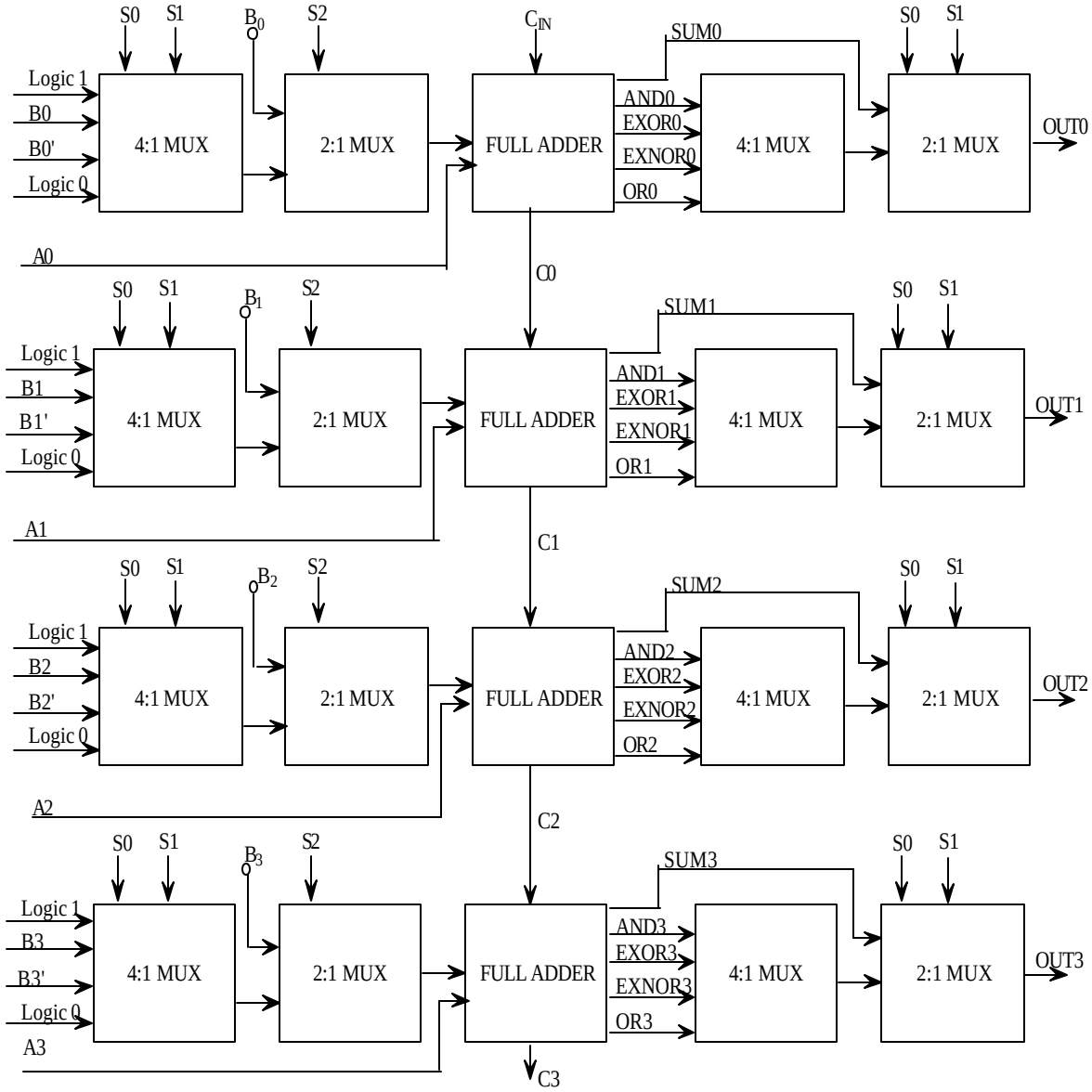


Figure 3.1: Block diagram of a 4-bit ALU.

simple and efficient in terms of area and timing. The pass transistor design reduces the parasitic capacitances and results in fast circuits.

There are two kinds of multiplexers implemented: 2 to 1 multiplexer and 4 to 1 multiplexer. Figure 3.2 shows the block diagram of a 4 to 1 MUX and Fig. 3.3 shows the circuit level diagram of the 4 to 1 MUX. Figure 3.4 shows the block diagram of a 2 to 1 MUX and Fig. 3.5 shows the circuit level diagram of the 2 to 1 MUX. The output of the multiplexer stage is passed as input to the full adder. A combination of the 2 to 1 MUX and 4 to 1 MUX at the input and output stage selects the signals depending on the operation being performed. Transmission gates select one of the inputs based on the value of the control signal. The input and select signals have been named as IN_n and S_n respectively, with the subscript n indicating the correct signal number. The input and the output stages have a combination of 2 to 1 multiplexer and 4 to 1 multiplexer to select one signal from a set of four signals. Figures 3.6 and 3.7 show how this logic has been implemented at input and output stage, respectively. The select signals are S_0, S_1 and S_2. Signal S_2 determines if the operation being performed is arithmetic or logical. The select signals S_0 and S_1 pick one of the four inputs or output signals and hence determine which of the four arithmetic or logical operations should be performed. For $S_2 = 0$, one of the four arithmetic operations is performed and for $S_2 = 1$, one of the four logical operations is performed. Table 3.1 shows the truth table for 4 to 1 MUX and Table 3.2 shows the truth table for 2 to 1 MUX.

3.1.2 Full Adder Design

In ALU, full adder forms the core of the entire design. The full adder performs the computing function of the ALU.

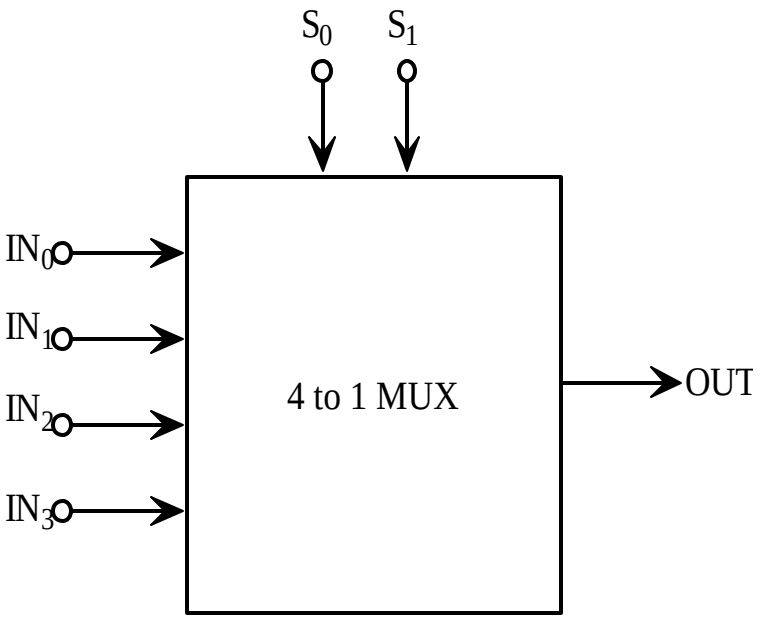


Figure 3.2: Block diagram of a 4 to 1 multiplexer.

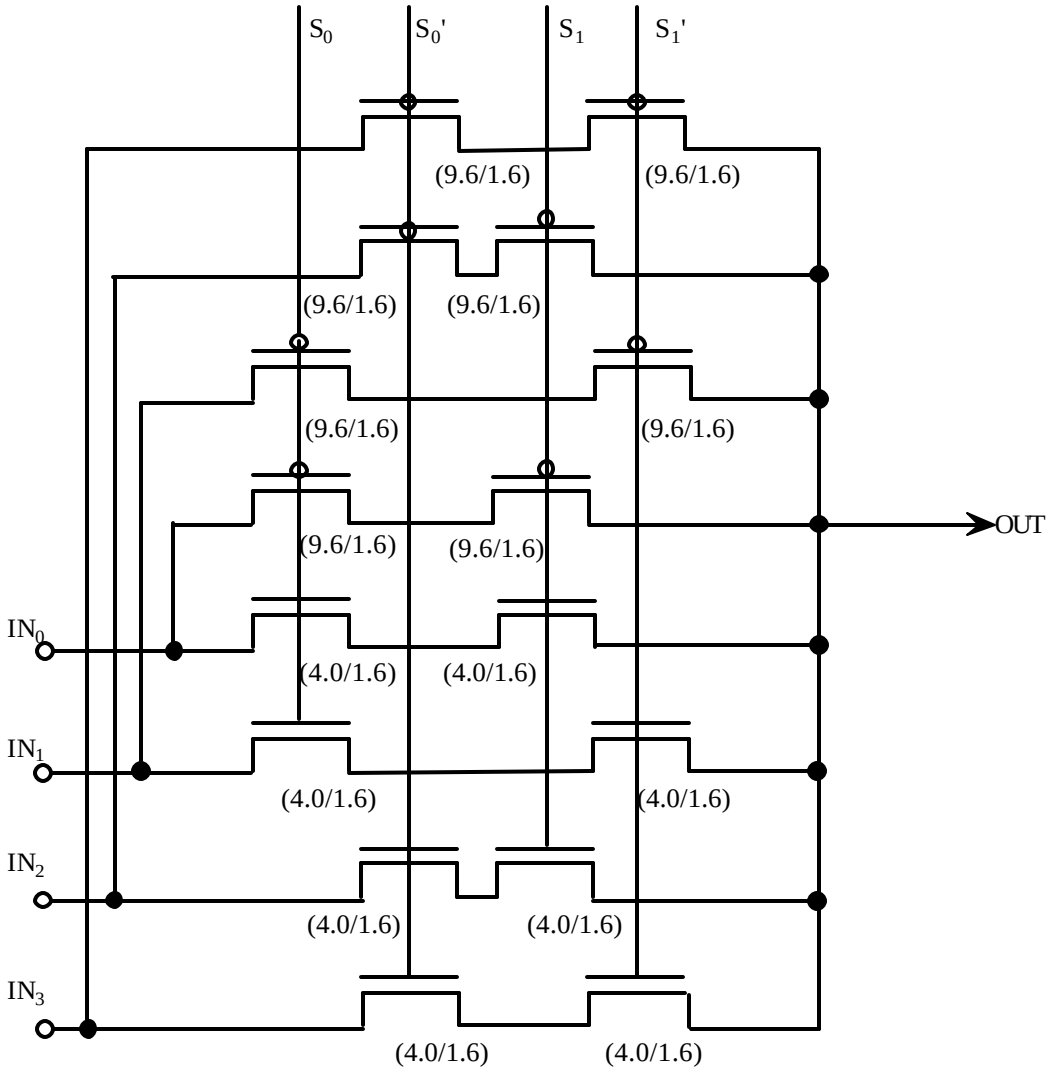


Figure 3.3: Circuit diagram of 4 to 1 multiplexer using pass transistors.

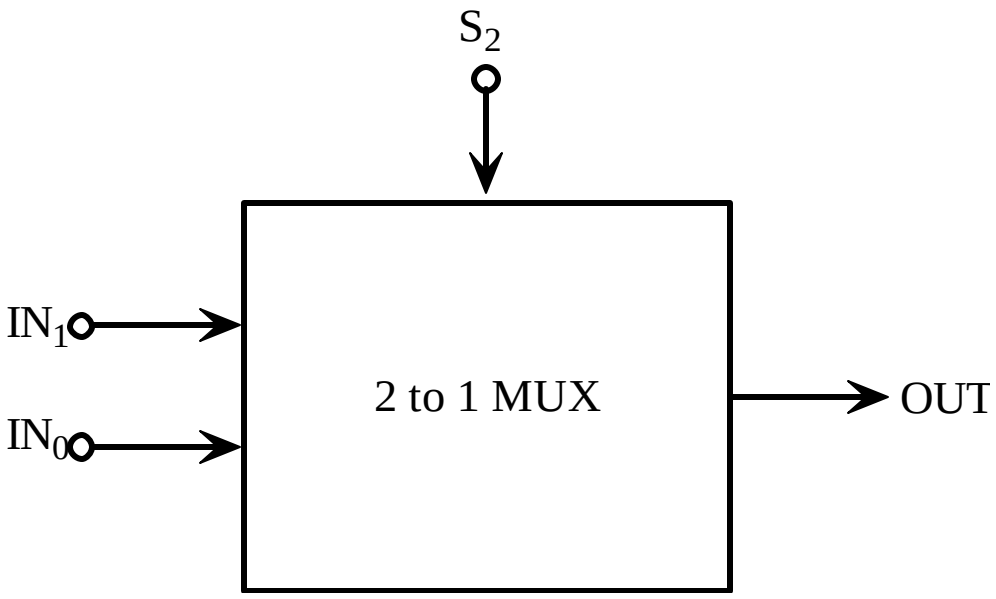


Figure 3.4: Block diagram of a 2 to 1 multiplexer.

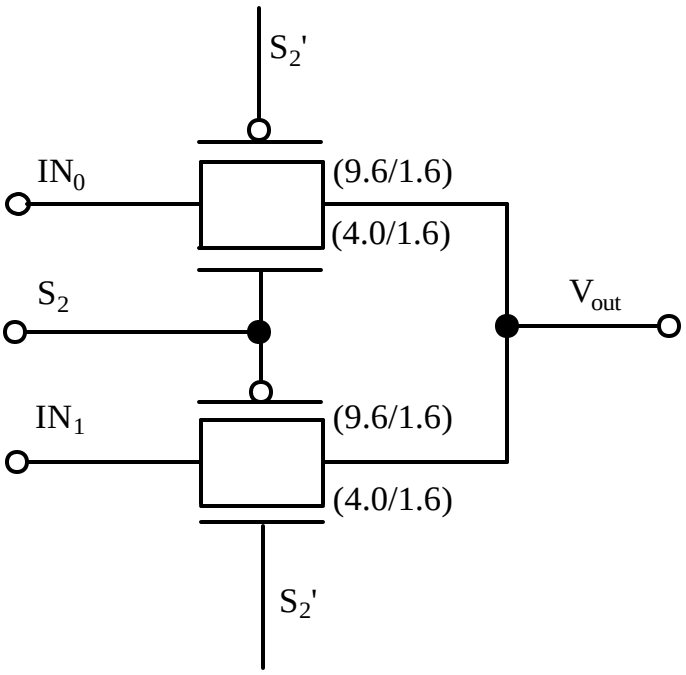


Figure 3.5: Circuit level diagram of a 2 to 1 multiplexer.

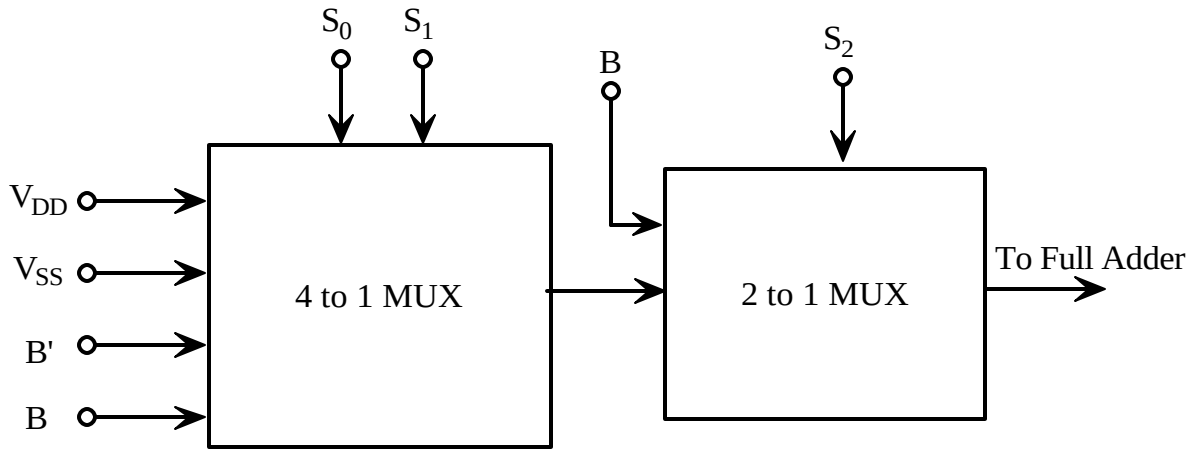


Figure 3.6: Block diagram of multiplexer logic at the input stage.

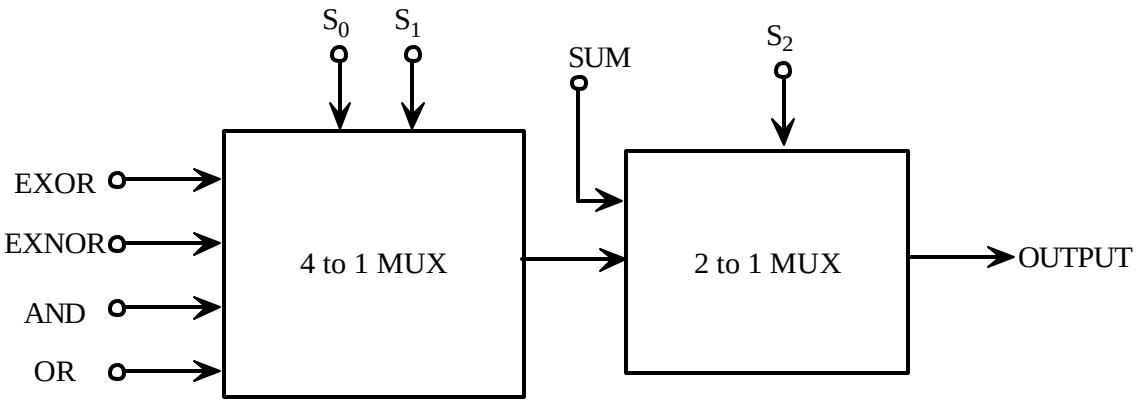


Figure 3.7: Block diagram of multiplexer logic at the output stage.

Table 3.1 Truth table of a 4 to 1 multiplexer

Select signal S_1	**Select signal S_2**	**Selected input**
0	0	IN_0
0	1	IN_1
1	0	IN_2
1	1	IN_3

Table 3.2 Truth table of a 2 to 1 multiplexer

Select signal S_2	**Operation performed**
0	Arithmetic
1	Logical

A full adder could be defined as a combinational circuit that forms the arithmetic sum of three input bits [38]. It consists of three inputs and two outputs. In our design, we have designated the three inputs as A, B and C_{IN}. The third input C_{IN} represents carry input to the first stage. The outputs are SUM and CARRY. Figure 3.8 shows the logic level diagram of a full adder. The Boolean expressions for the SUM and CARRY bits are as shown below.

$$SUM = A \oplus B \oplus C_{IN} \tag{3.1}$$

$$CARRY = A \bullet B + A \bullet C_{IN} + B \bullet C_{IN} \tag{3.2}$$

SUM bit is the EXOR function of all three inputs and CARRY bit is the AND function of the three inputs. The truth table of a full adder is shown in Table 3.3. The truth table also indicates the status of the CARRY bit; that is to say, if that carry bit has been generated or deleted or propagated. Depending on the status of input bits A and B, the CARRY bit is either generated or deleted or propagated [8]. If either one of A or B inputs is '1', then the previous carry is just propagated, as the sum of A and B is '1'. If both A and B are '1's then carry is generated because summing A and B would make output SUM '0' and CARRY '1'. If both A and B are '0's then summing A and B would give us '0' and any previous carry is added to this SUM making CARRY bit '0'. This is in effect deleting the CARRY. To construct an n-bit adder we have to cascade n such 1-bit adders. We have used this ripple carry adder (RCA) configuration in our ALU design.

In RCA, the CARRY bit ripples all the way from first stage to n^{th} stage. Figure 3.9 shows the block diagram of a four-bit ripple carry adder. The delay in a RCA depends on the number of stages cascaded and also the input bits' patterns.

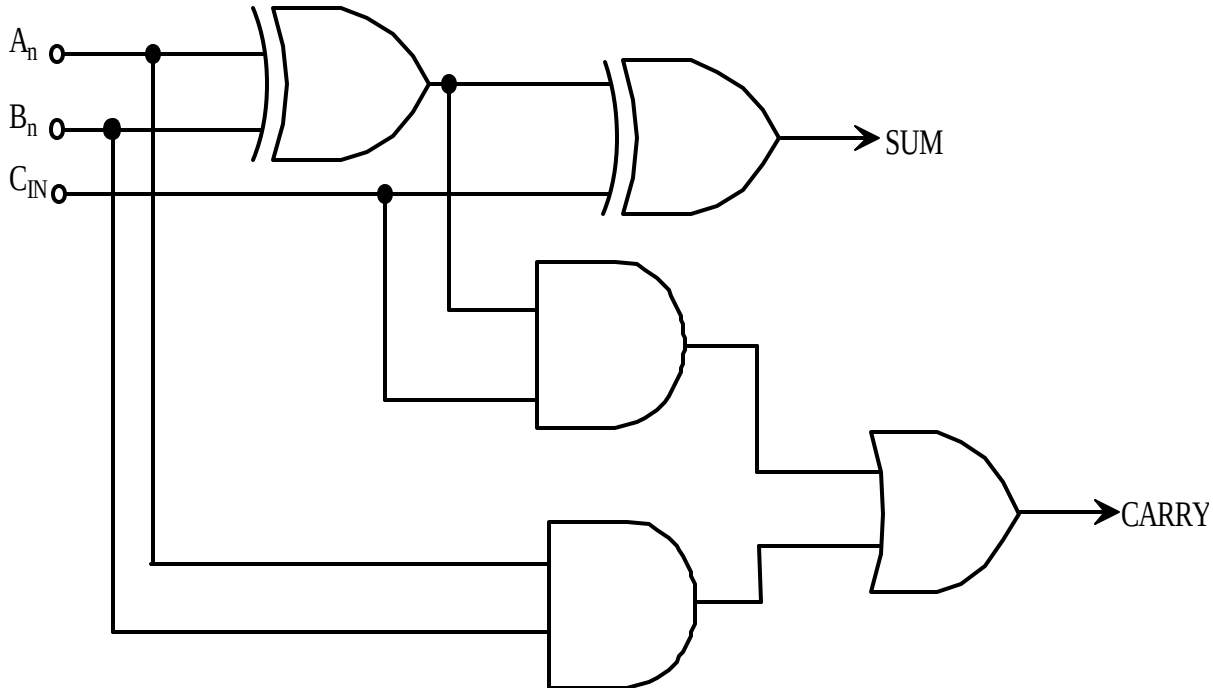


Figure 3.8: Logic level diagram of a full adder.

Table 3.3 Truth table of a full adder

A	**B**	**C_{IN}**	**SUM**	**CARRY**	**Carry Status**
0	0	0	0	0	Delete
0	0	1	1	0	Delete
0	1	0	1	0	Propagate
0	1	1	0	1	Propagate
1	0	0	1	0	Propagate
1	0	1	0	1	Propagate
1	1	0	0	1	Generate
1	1	1	1	1	Generate

For certain input patterns, a CARRY is neither generated nor propagated. This way the CARRY bit need not ripple through the stages. This effectively reduces the delay in the circuit. On the other hand, certain input patterns generate carry bit in the first stage itself, which might have to ripple through all the stages. This definitely increases the delay in the circuit. The propagation delay of such a case, also called critical path, is defined as worst-case delay over all possible input patterns. In a ripple carry adder, the worst-case delay occurs when a carry bit propagates all the way from least significant bit position to most significant bit position. The total delay of the adder would be an addition of delay of a SUM bit and delay of a CARRY bit multiplied by number of bits minus one in the input word, given by Eq. (3.3) [37, 38].

$$T_{adder} = (N-1)\, T_{carry} + T_{sum} \quad (3.3)$$

Where N is number of bits in input word, T_{carry} and T_{sum} are propagation delays from one stage to another. For an efficient ripple carry adder, it is important to reduce T_{carry} than T_{sum} as the former influences the total adder delay more.

In our design, we have implemented the full adder design using MIFG CMOS devices. Essentially the SUM bit of a full adder involves building of XOR gates to realize its function and similarly CARRY bit needs AND gates to realize its Boolean expression. The logic function of XOR gate can be realized using MIFG CMOS transistors. The full adder constructed from MIFG devices uses only eight transistors, four MIFG transistors and four conventional transistors. Figure 3.10 shows the XOR logic in MIFG CMOS, required for generating the SUM output of the full adder. The unit size capacitance is 500 fF and other capacitors are integral multiples of the unit size capacitor.

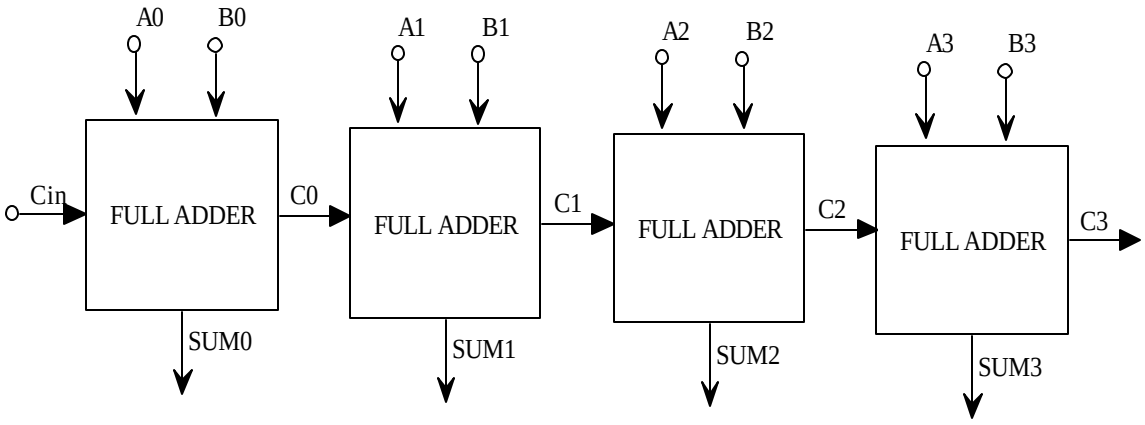


Figure 3.9: Block diagram of a 4-bit ripple carry adder.

Inverters #1 and #2 are MIFG CMOS inverters and inverter #3 is a standard inverter required to complement the output from the MIFG inverters #1 and #2. The inputs to the first MIFG inverter are A, B and C_{IN}. MIFG CMOS inverter #1 acts as the pre-gate input to inverter #2. The voltage on the floating gate of inverter #1 is determined by the following equation,

$$\Phi_F = \frac{C(V_A + V_B + V_{CIN})}{3C + C_0 + C_p} \tag{3.4}$$

Where V_A, V_B and V_C are input voltages. C is the coupling capacitance for each input and C_0 is the gate oxide capacitance and C_p is the parasitic capacitance. From extracted netlists of trial designs, we approximated the sum of gate oxide and parasitic capacitances to be nearly 250 fF. According to the principle of operation of the MIFG CMOS inverter, the output of inverter #1, V_1, changes state when Φ_F goes above the threshold value of the device. The threshold voltage, V_{th} is approximately taken as 1.0V for n-MOSFET. Equations (3.5-3.8) show the calculation of Φ_F for three exclusive combinations of inputs A, B and C_{IN} using Eq. (3.4). The input voltage signals are set at 3V for logic '1' and 0V for logic '0'.

$$\Phi_F = \frac{500\times(0+0+0)+500\times 0}{3\times 500+250} = 0\text{V} \tag{3.5}$$

so $\Phi_F < V_{th}$. Hence V_1 is HIGH (logic '1').

$$\Phi_F = \frac{500\times(0+0+3)+500\times 0}{3\times 500+250} = 0.85\text{V} \tag{3.6}$$

so $\Phi_F < V_{th}$. Hence V_1 is HIGH (logic '1').

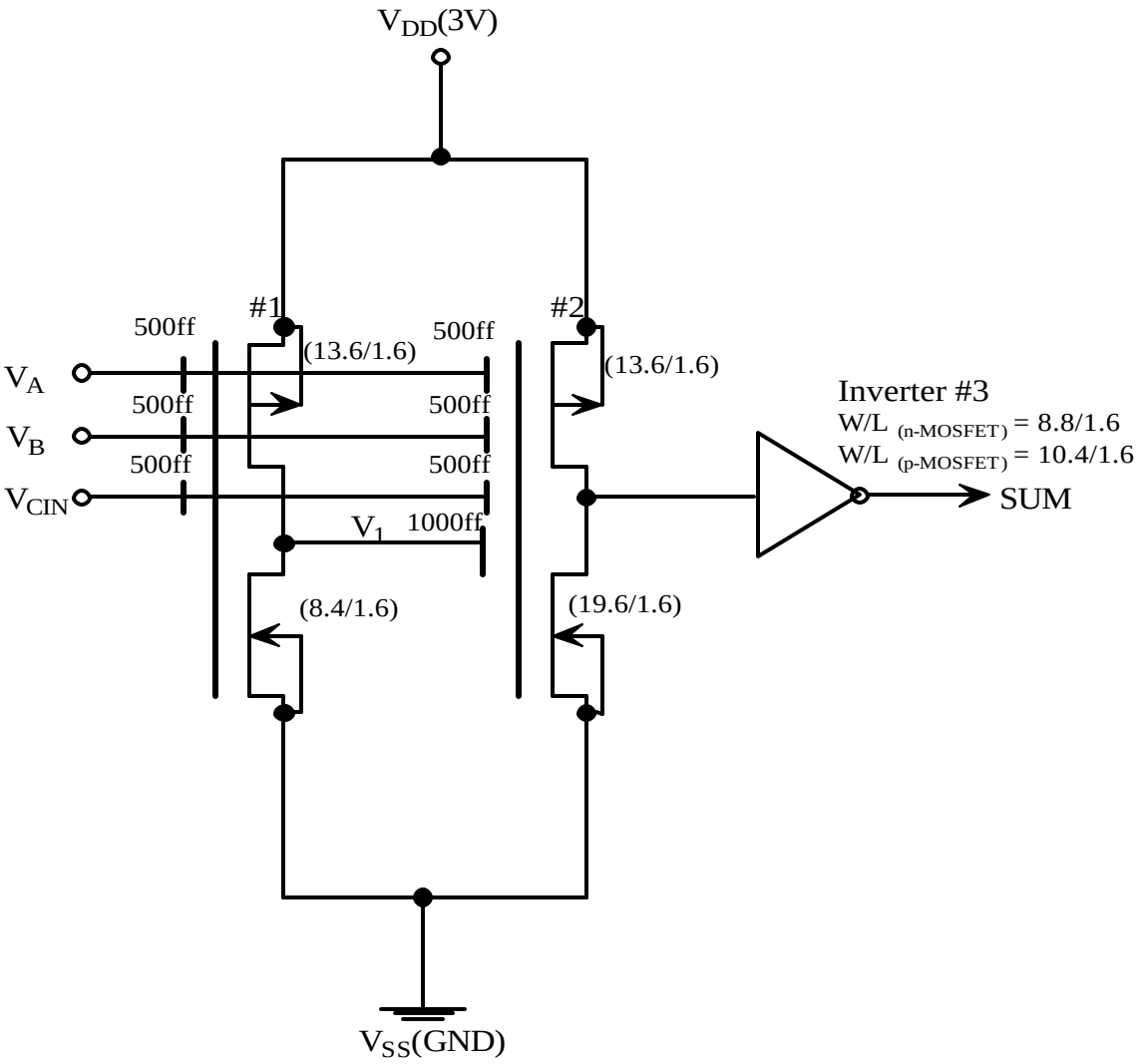


Figure 3.10: XOR gate using MIFG CMOS inverters to generate SUM bit.

$$\Phi_F = \frac{500\times(0+3+3)+500\times 0}{3\times 500+250} = 1.7\text{V} \quad (3.7)$$

so $\Phi_F > V_{th}$. Hence V_1 is LOW (logic '0').

$$\Phi_F = \frac{500\times(3+3+3)+500\times 0}{3\times 500+250} = 2.5\text{V} \quad (3.8)$$

so $\Phi_F > V_{th}$. Hence V_1 is LOW (logic '0').

The output, V_1 is the input for MIFG inverter #2 along with inputs are A, B and C_{IN} . The value of Φ_F for the inverter #2 is computed by the following equation,

$$F_F = \frac{C\times V_A + C\times V_B + C\times V_{CIN} + 2CV_1}{5C + C_0 + C_p} \quad (3.9)$$

The output of MIFG inverter #2 should go LOW (logic '0') only when odd number of inputs is HIGH (logic '1') to realize an XOR function. Hence weight of capacitor coupling V_1 to MIFG inverter #2 should be decided accordingly. We set the weight of coupling capacitor for input V_1 to 2C in the second stage. This allows for the floating gate voltage Φ_F of MIFG inverter #2 to be pulled LOW (logic '0') even when just one of the inputs is HIGH (logic '1'). This is shown in Eqs. (3.10-3.13) below. The output V_1, however, is not completely pulled up to V_{DD} (3V). It is degraded to around 1.9V. This has been addressed as one of the design issues in the second chapter, pages 37-39. Substituting the values for voltage V_1 from Eqs. (3.5-3.8), the value of Φ_F is calculated for exclusive combinations of inputs A, B, C_{in}, using equation 3.9. The calculations are shown as follows,

$$F_F = \frac{500\times(0+0+0+2\times 1.9)}{5\times 500+550} = 0.6\text{V} \quad (3.10)$$

so $\Phi_F < V_{th}$. Hence SUM is LOW (logic '0').

$$F_F = \frac{500\times(0\ +0+3+2\times1.9)}{5\times500+550} = 1.10\text{V} \tag{3.11}$$

so $\Phi_F > V_{th}$. Hence SUM is HIGH (logic '1').

$$F_F = \frac{500\times(0\ +3+3+2\times0)}{5\times500+550} = 0.98\text{V} \tag{3.12}$$

so $\Phi_F < V_{th}$. Hence SUM is LOW (logic '0').

$$F_F = \frac{500\times(3\ +3+3+2\times0)}{6\times500+550} = 1.5\text{V} \tag{3.13}$$

so $\Phi_F > V_{th}$. Hence SUM is HIGH (logic '1').

Table 3.4 gives the truth table for the SUM bit from Fig. 3.10 and also shows the value of Φ_F compared to threshold voltage, V_{th}, for each of the inputs bits' combination. Figure 3.11 shows the AND logic required for generating the CARRY output of the full adder. Each input capacitor in Fig. 3.11 is a unit size capacitance of 500ff. Inverter #1 is a MIFG CMOS inverter and inverter #4 is a standard inverter required to complement the output from the MIFG inverter. The inputs to the MIFG inverter are A, B and C_{IN}. The voltage on the floating gate of inverter #1 is determined by the following equation,

$$\Phi_F = \frac{C(V_A + V_B + V_{CIN})}{3C + C_0 + C_p} \tag{3.14}$$

Equations (3.15-3.17) show the calculation of Φ_F for three exclusive combinations of inputs are A, B and C_{IN} , using Eq. (3.14) and hence the status of output CARRY

$$\Phi_F = \frac{500\times(0\ +0+0)}{3\times500+250} = 0\text{V} \tag{3.15}$$

so $\Phi_F < V_{th}$. Hence CARRY is LOW (logic '0').

$$\Phi_F = \frac{500\times(0+0+3)}{3\times500+250} = 0.85\text{V} \quad (3.16)$$

so $\Phi_{F<} V_{th}$. Hence CARRY is LOW (logic '0').

$$\Phi_F = \frac{500\times(0+3+3)}{3\times500+250} = 1.7\text{V} \quad (3.17)$$

so $\Phi_{F>} V_{th}$. Hence CARRY is HIGH (logic '1').

$$\Phi_F = \frac{500\times(3+3+3)}{3\times500+250} = 2.5\text{V} \quad (3.18)$$

so $\Phi_{F>} V_{th}$. Hence CARRY is HIGH (logic '1').

Table 3.5 shows the truth table generated by inverter pair #1 and #4 from Fig. 3.11 for the CARRY bit. The value of Φ_F is compared with threshold voltage V_{th}. The combined logic for SUM and CARRY forming a complete full adder design is shown in Fig. 3.12. To realize a 2-input OR function, additional circuitry using MIFG MOSFETs is added to the full adder circuit. Figure 3.13 shows the additional circuit needed to realize a 2-input OR gate. This table is similar to the truth table of an OR gate. In Fig. 3.13, the voltage on the floating gate of inverter #5 is determined by the equation;

$$F_F = \frac{C(V_A+V_B) + C\times V_{DD}}{3C+C_0+C_p} \quad (3.19)$$

Voltage on floating gate of inverter #5 goes above threshold voltage whenever one of the input bits goes HIGH. Inverter #6 complements the output of the MIFG inverter #5 to perform OR operation on inputs A and B. In the design of OR gate, we need the floating gate voltage Φ_F to go higher than threshold voltage whenever at least one input goes HIGH. To realize this design with given MOSFET parameters, we see the need to precharge the floating gate to a certain voltage level. This is done by adding a unit capacitor which is connected permanently to V_{DD} .

Table 3.4 Truth table generated by inverters 1, 2 and 3 for SUM bit

V_A	V_B	V_{CIN}	V_1	F $_F$ V_{th}	**Sum-bit: inverter #3 output**
0	0	0	1	$\Phi_F = 0.6 < V_{th}$	0
0	0	1	1	$\Phi_F = 1.1 > V_{th}$	1
0	1	0	1	$\Phi_F = 0.98 < V_{th}$	1
0	1	1	0	$\Phi_F = 0.98 < V_{th}$	0
1	0	0	1	$\Phi_F = 1.1 > V_{th}$	1
1	0	1	0	$\Phi_F = 0.98 < V_{th}$	0
1	1	0	0	$\Phi_F = 0.98 < V_{th}$	0
1	1	1	0	$\Phi_F = 1.5 > V_{th}$	1

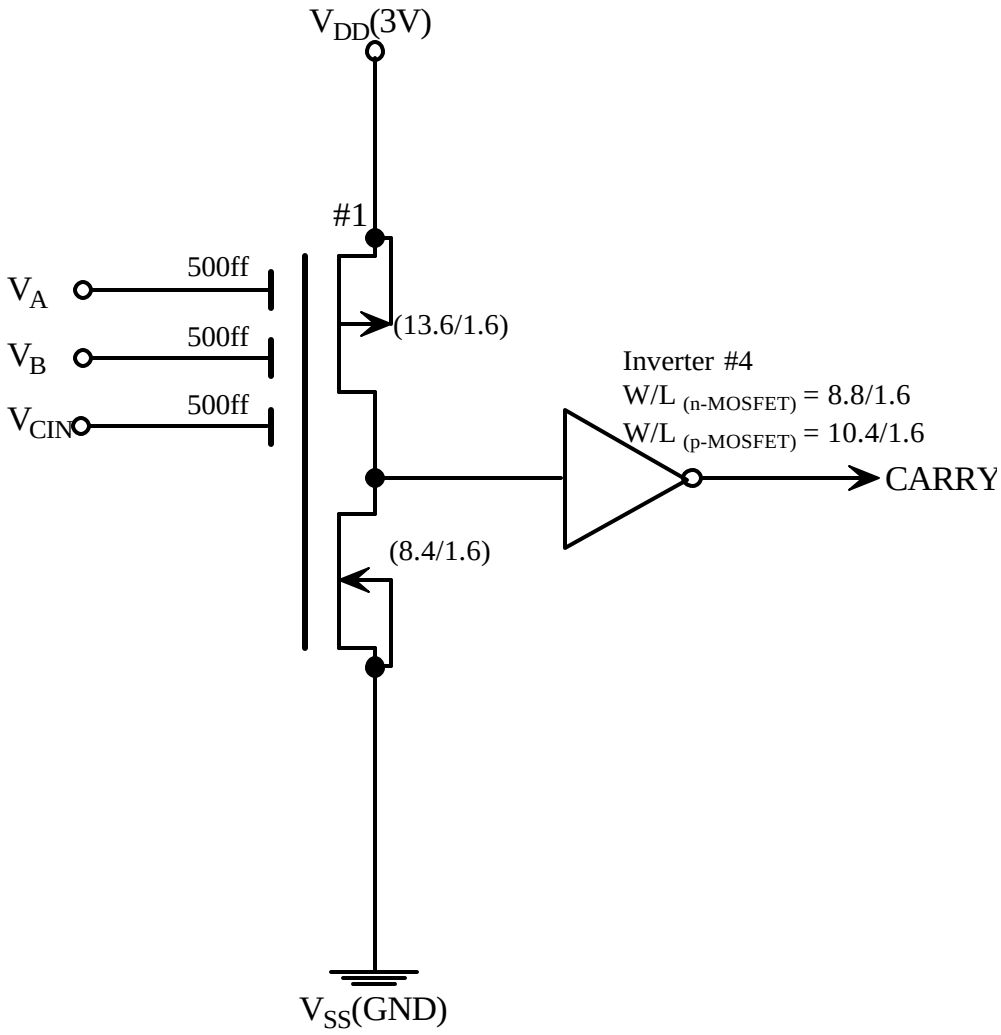


Figure 3.11: AND gate using MIFG CMOS inverter to generate CARRY bit.

Table 3.5 Truth table generated by inverters #1 and #4 for CARRY bit.

V_A	V_B	V_{CIN}	V_1	F $_F$ V_{th}	**Carry-bit: inverter #4 output**
0	0	0	1	$\Phi_F = 0 < V_{th}$	0
0	0	1	1	$\Phi_F = 0.85 < V_{th}$	1
0	1	0	1	$\Phi_F = 085 < V_{th}$	1
0	1	1	0	$\Phi_F = 1.7 > V_{th}$	0
1	0	0	1	$\Phi_F = 0.85 < V_{th}$	1
1	0	1	0	$\Phi_F = 1.7 > V_{th}$	0
1	1	0	0	$\Phi_F = 1.7 > V_{th}$	0
1	1	1	0	$\Phi_F = 2.5 > V_{th}$	1

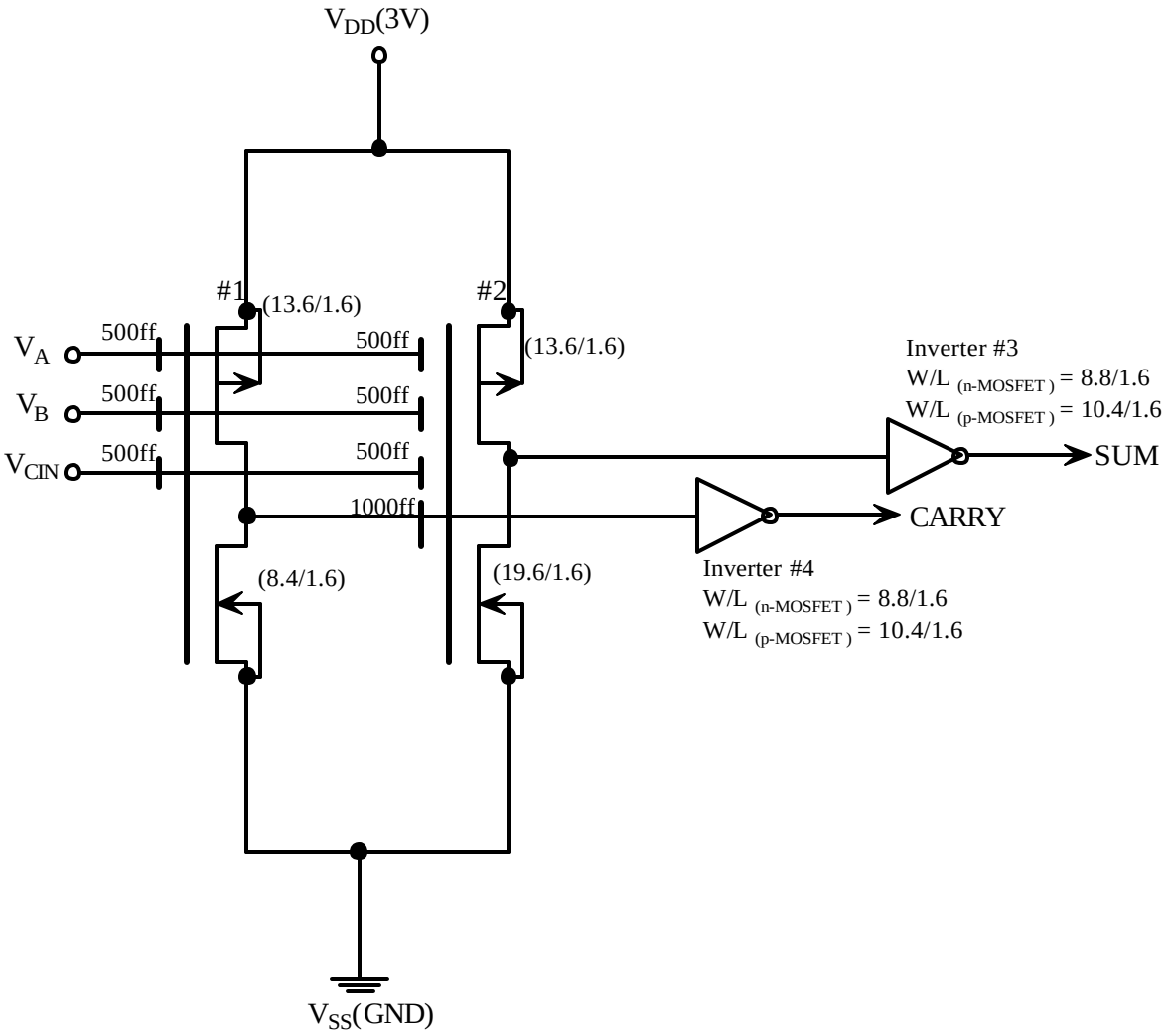


Figure 3.12: Full adder design using MIFG CMOS transistors.

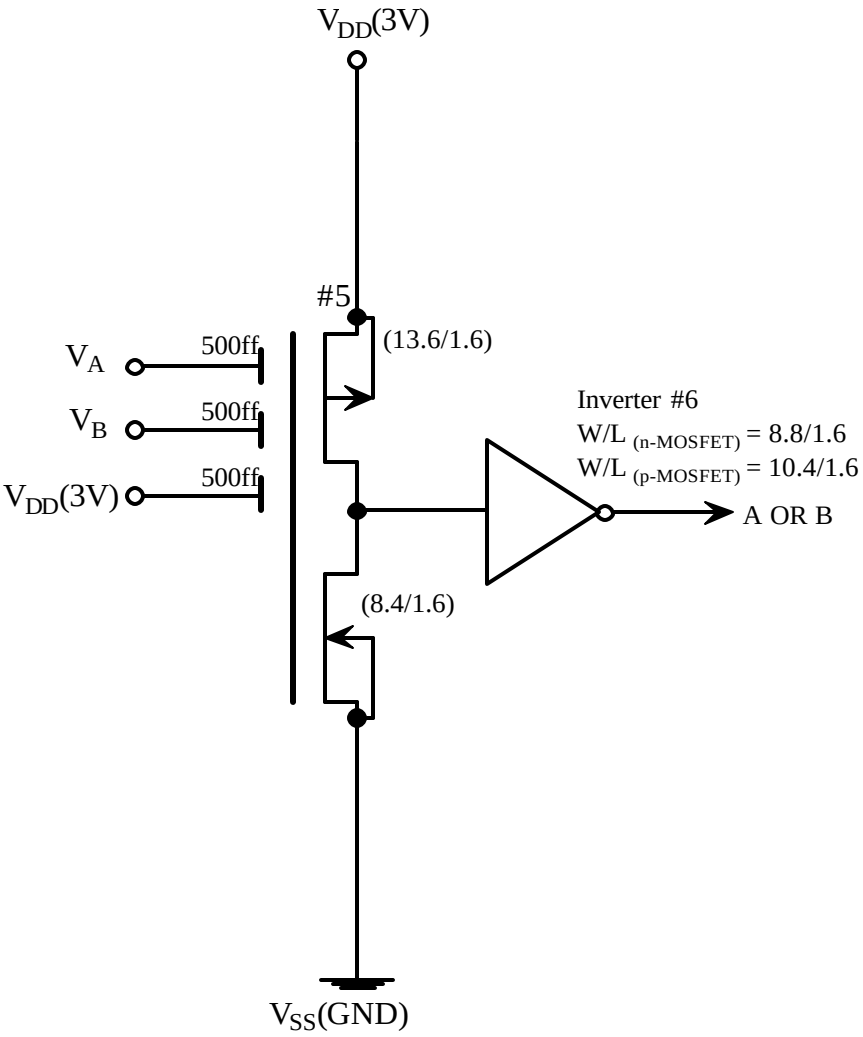


Figure 3.13: OR gate using MIFG CMOS inverters.

This makes the floating gate of inverter #5 sensitive to small changes in voltage on it. So even if one input goes HIGH, Φ_F pulls the output of the MIFG inverter to LOW. Equations (3.20-3.22) show the calculations for OR output bit using Eq. (3.19), for all exclusive combinations of inputs A and B.

$$F_F = \frac{500\times(0+0)+500\times 3}{3\times 500+200} = 0.8\text{V} \quad (3.20)$$

so $\Phi_{F<} V_{th}$. Hence OR bit is LOW (logic '0').

$$F_F = \frac{500\times(0+3)+500\times 3}{3\times 500+200} = 1.8\text{VV} \quad (3.21)$$

so $\Phi_{F>} V_{th}$. Hence OR bit is HIGH (logic '1').

$$F_F = \frac{500\times(3+3)+500\times 3}{3\times 500+200} = 2.6\text{V} \quad (3.20)$$

so $\Phi_{F>} V_{th}$. Hence OR bit is HIGH (logic '1').

Table 3.6 shows how the output bit is obtained by variation of Φ_F for different combinations of input bits and comparison of Φ_F with threshold voltage V_{th}.

The full adder design along with the additional OR gate logic comprises of twelve transistors as shown in Fig. 3.14. This circuit put together performs four arithmetic functions; ADD, SUBTRACT, INCREMENT and DECREMENT and four logical operations; EXOR, EXNOR, AND and OR.

3.1.3 ALU Design

Each stage of the 4-bit ALU is comprised of the full adder block and the multiplexer block at the input and output stages. Figure 3.15 shows the block diagram of the ALU for all four stages. Each stage of the ALU has five inputs given to it; Logic '1', Logic '0', A, B and complement of B (B'). Logic '1' and Logic '0' are realized by tying

Table 3.6 Truth table generated by inverter pair #5 and #6 for the OR gate

V_A	V_B	F $_F$ V_{th}	**OR output: inverter #6 output**
0	0	$\Phi_F = 0.8 < V_{th}$	0
0	1	$\Phi_F = 1.8 > V_{th}$	1
1	0	$\Phi_F = 1.8 > V_{th}$	1
1	1	$\Phi_F = 2.6 > V_{th}$	1

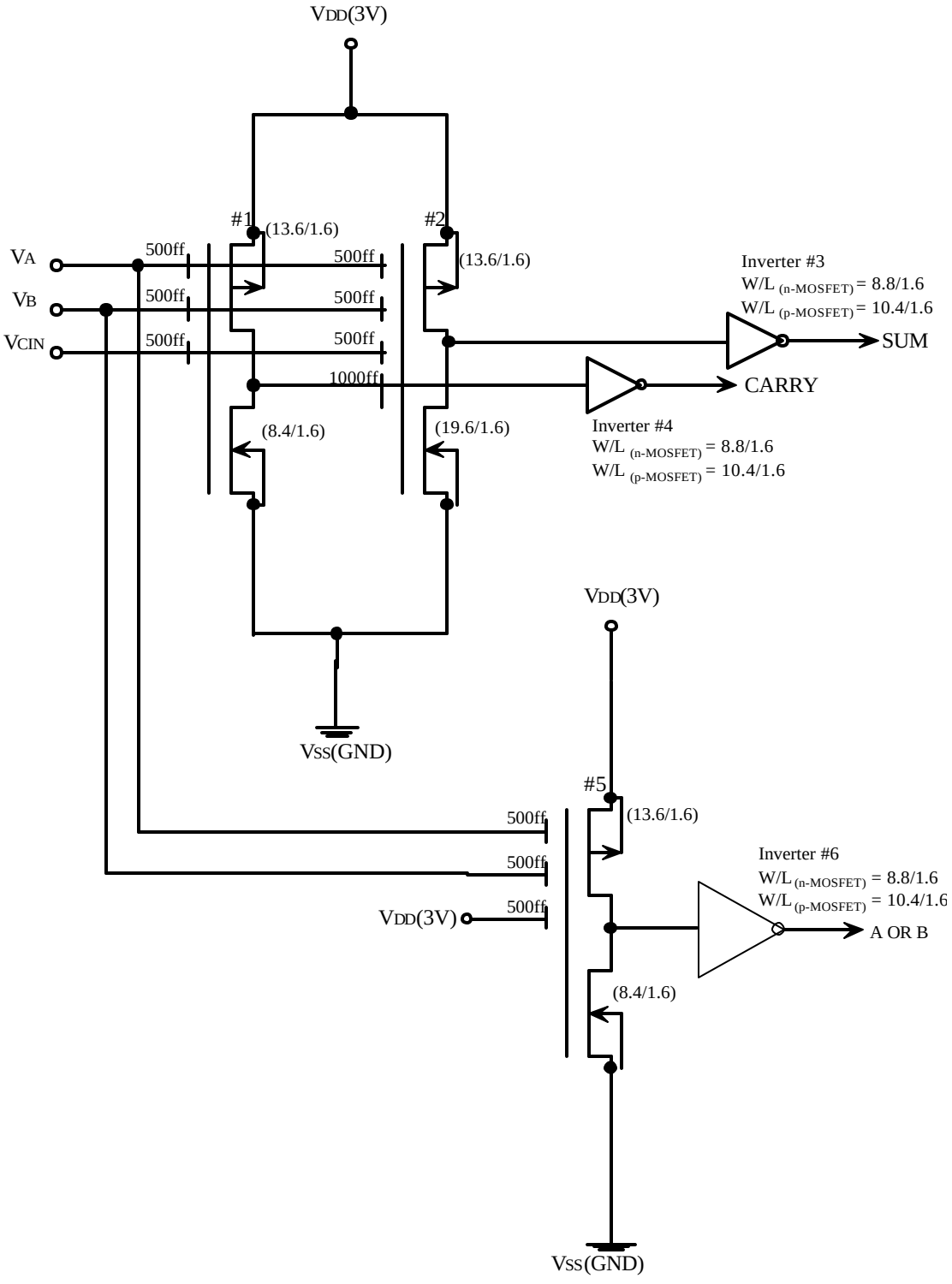


Figure 3.14: Full adder design using MIFG CMOS transistors for arithmetic and logic operations.

the inputs to V_{DD} and V_{SS}, respectively in Fig. 3.15. The inputs, Logic '1' and Logic '0' are used for the INCREMENT and DECREMENT operations respectively. The complement of B is used for SUBTRACTION operation. The full adder performs the SUBTRACT operation by two's complement method. An INCREMENT operation is analyzed as adding a '1' to the addend and DECREMENT is seen as a subtraction operation. The outputs from the full adder are SUM, EXOR, EXNOR, AND and OR. Based on the condition of the select signals, the multiplexer stage selects the appropriate inputs and gives it to the full adder. The full adder computes the results. The multiplexer at the output stage selects the appropriate output and sends it out. Table 3.7 shows the truth table for the operations performed by the ALU based on the status of the select signals.

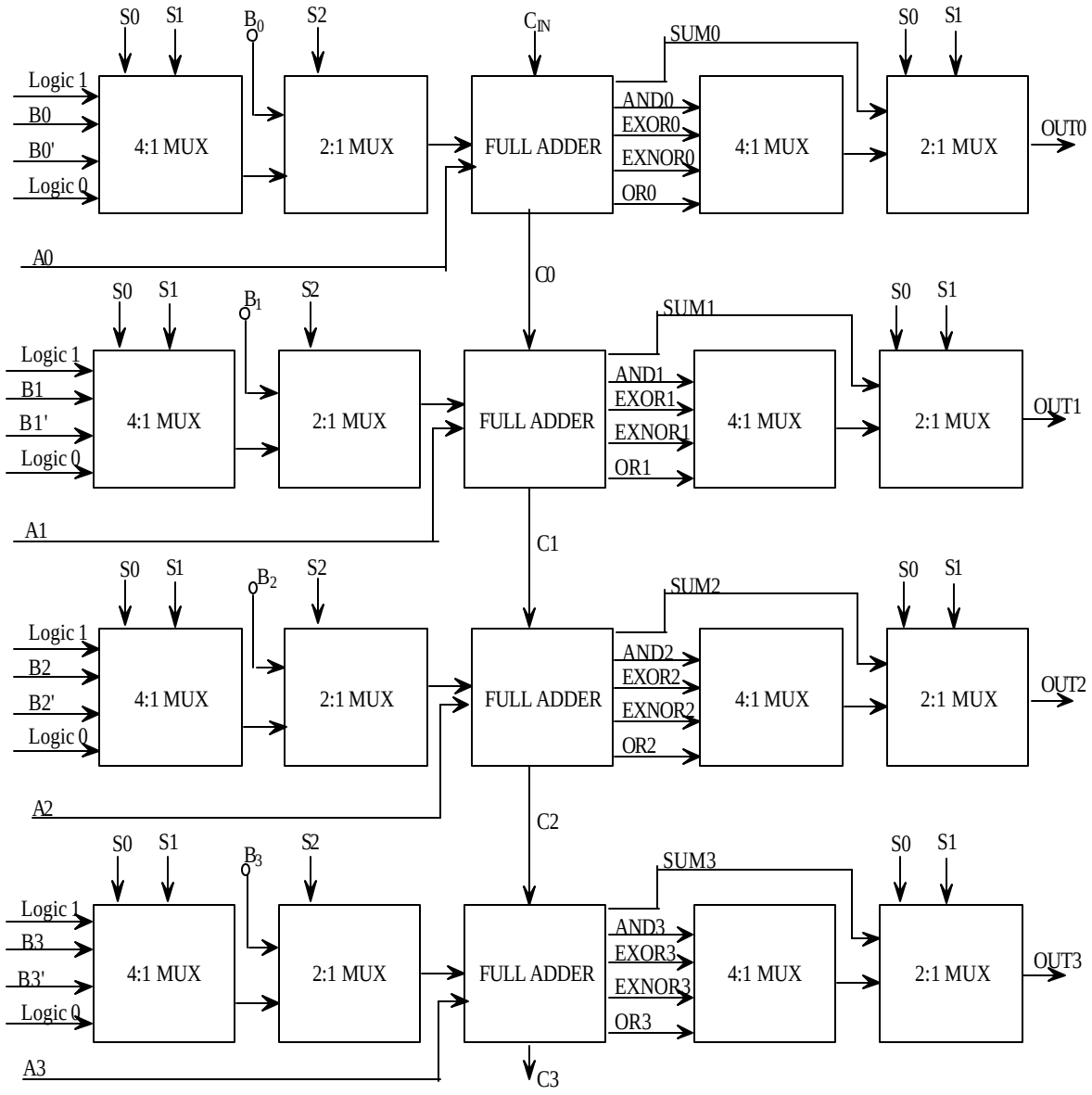


Figure 3.15: Block diagram of a 4-bit ALU.

Table 3.7 Truth table for the 4-bit ALU.

S_2	S_1	S_0	**Operation performed**
0	0	0	INCREMENT
0	0	1	DECREMENT
0	1	0	ADDITION
0	1	1	SUBTRATION
1	0	0	AND
1	0	1	OR
1	1	0	EXOR
1	1	1	EXNOR

Chapter 4

Design Issues, Simulations and Experimental Data

This chapter discusses the simulation and design issues associated with our ALU. The experimental data has also been presented. The 4-bit ALU is designed using Tanner L-Edit 8.03 tool in standard 1.5µm CMOS technology. The design is simulated using MicroSim Pspice tool. The SPICE level 3 MOS model parameters was used for pre-layout and post-layout simulations. The level 3 MOS model parameters are listed in Appendix A.

4.1 Multiplexer

The PMOS transistors in 4:1 multiplexer and 2:1 multiplexer have been designed with a W/L ratio of 9.6/1.6 and NMOS transistors with a W/L ratio of 4.0/1.6. The width of PMOS transistors in multiplexers have been increased to reduce rise and fall times. Figure 4.1 shows the layout of a 2 to 1 MUX. Figure 4.2 shows the layout of a 4 to 1 MUX. The MUX output, selected from inputs IN_0, IN_1, IN_2 and IN_3, depending on the status of select signals S_0 and S_1 is tabulated in Table 4.1. The 4 to 1 MUX designed was simulated by testing it for various combinations of select signals shown in Fig. 4.3. IN_0 is an input pulse of period 200ns, IN_1 is a constant voltage of 3V, IN_2 is complement of pulse IN_0 and IN_3 is a constant voltage of 0V. Table 4.2 shows time period and status of select signals S_0 and S_1 and output, V_{out} based on the select signals. In this figure, the output waveform V_{out} follows one of the four inputs depending on status of select signals S_0 and S_1.

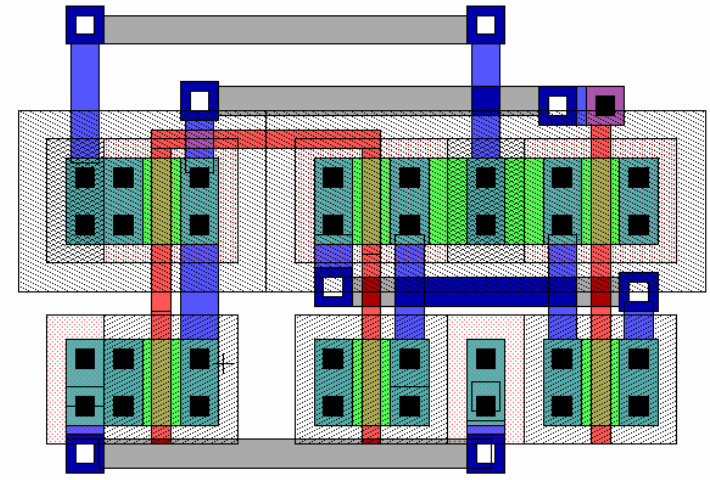


Figure 4.1: Layout of a 2 to 1 MUX.

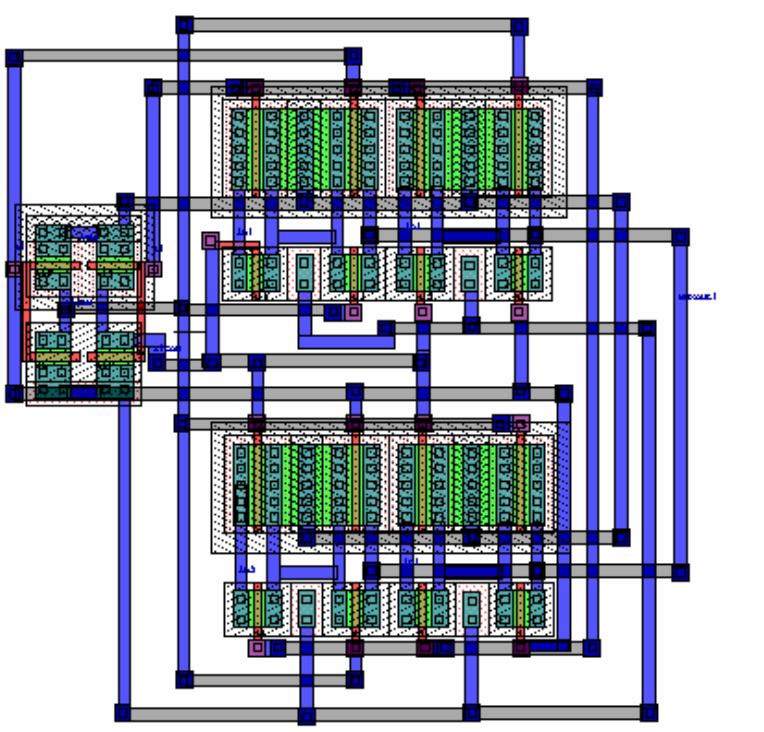


Figure 4.2: Layout of a 4 to 1 MUX.

Table 4.1 Truth table of a 4 to 1 multiplexer

Select signal S_1	**Select signal S_2**	**Selected input**
0	0	IN_0
0	1	IN_1
1	0	IN_2
1	1	IN_3

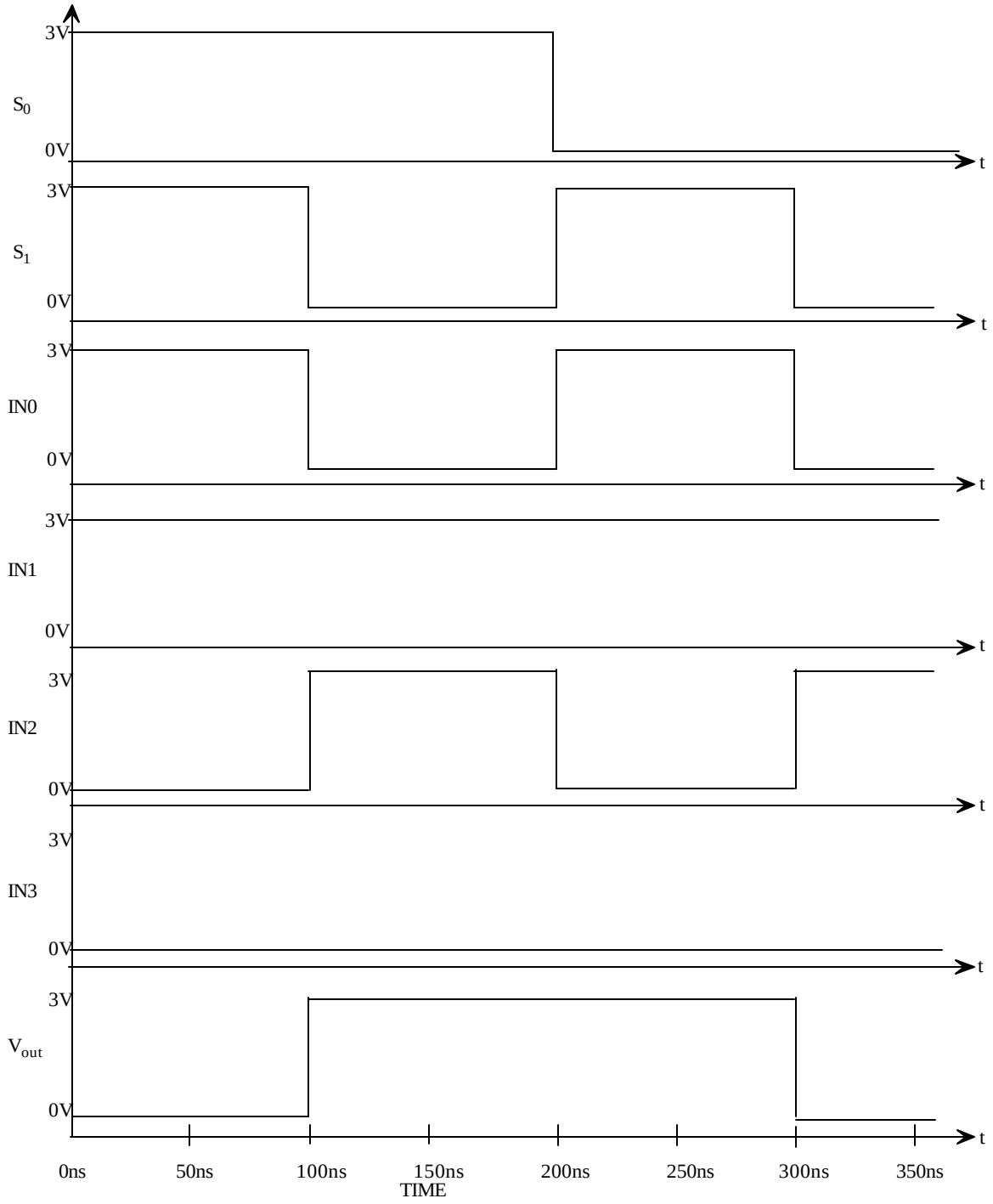


Figure 4.3: 4 to 1 MUX output waveforms for all combinations of select signals S_0 and S_1.

Table 4.2 Time period and status of select signals S_0 and S_1 and value output V_{out} based on the select signals

Time duration	**Select signals status**	**Output of MUX**
0-100ns	$S_0 = 1$ $S_1 = 1$	$V_{out} = IN_3$ (‘0’)
100-200ns	$S_0 = 1$ $S_1 = 0$	$V_{out} = IN_2$(‘1’)
200-300ns	$S_0 = 0$ $S_1 = 1$	$V_{out} = IN_1$(‘1’)
300-400ns	$S_0 = 0$ $S_1 = 0$	$V_{out} = IN_0$(‘0’)

4.2 Full Adder

The MIFG inverter has been realized using first polysilicon for the floating gate and input voltages are coupled via capacitors to the gate using second polysilicon contacts. Figure 4.4 shows a MIFG CMOS full adder circuit. In Fig. 4.4, inverters #1 and #2 are MIFG CMOS inverters and inverters #3 and #4 are conventional CMOS inverters. The inputs are coupled capacitively to the MIFG inverters. When MIFG inverter #1 and #2 are cascaded, input capacitors of MIFG inverter #2 act as load for inverter #1. From the SPICE netlist, the parasitic capacitances are in the range of 100fF-400fF. These input capacitors also add up to the parasitic capacitance already present on the floating gates of MIFG inverters. These capacitances cause rise and fall times of input signals to increase. Slow rise and fall times cause delay in the output waveforms. It is therefore necessary to resize the transistors by increasing their W/L ratios. The W/L ratios of MIFG inverter #1 for PMOS and NMOS transistors are 13.6/1.6 and 8.4/1.6, respectively. In the cascaded stage, MIFG inverter #2 has a large parasitic capacitance of 300fF on its gate. This causes increase in the rise and fall times of the input waveform. W/L ratio of the NMOS transistor has been increased to 19.6/1.6 to reduce the fall time of the output of inverter #2, which is the invert of SUM bit. However, the outputs of the MIFG inverters #1 and #2 still have slow rise and fall times. The result of this is, the SUM and CARRY bit waveforms have slow rise and fall times. To reduce these transition times, the W_p/W_n ratios of the conventional inverters #3 and #4 were set to 10.4/8.8. The layout of the full adder is shown in Fig. 4.5. The full adder design occupies an area of 293 x 160μm^2.

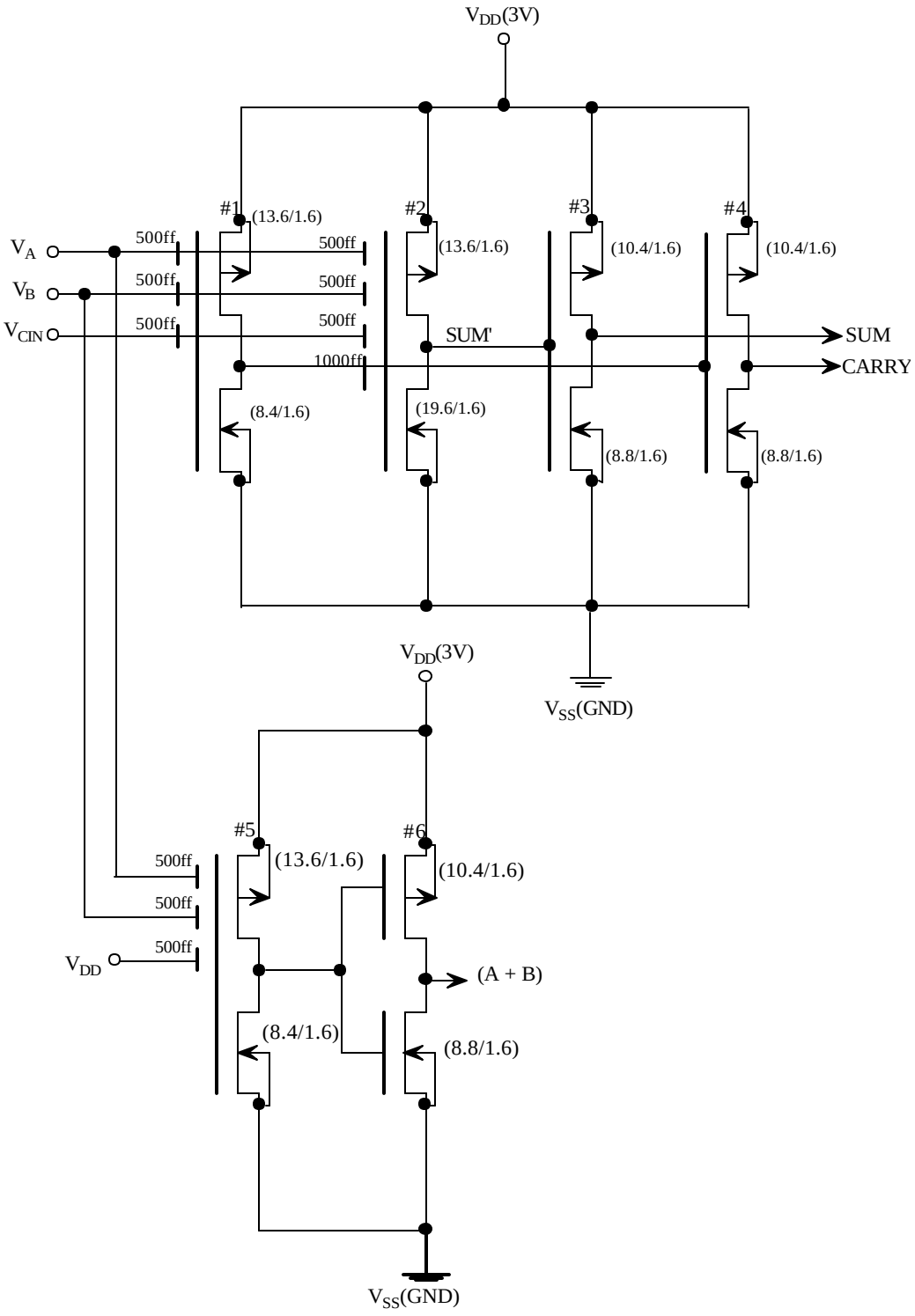


Figure 4.4: Circuit diagram of a full adder, with additional OR logic, using MIFG CMOS inverters.

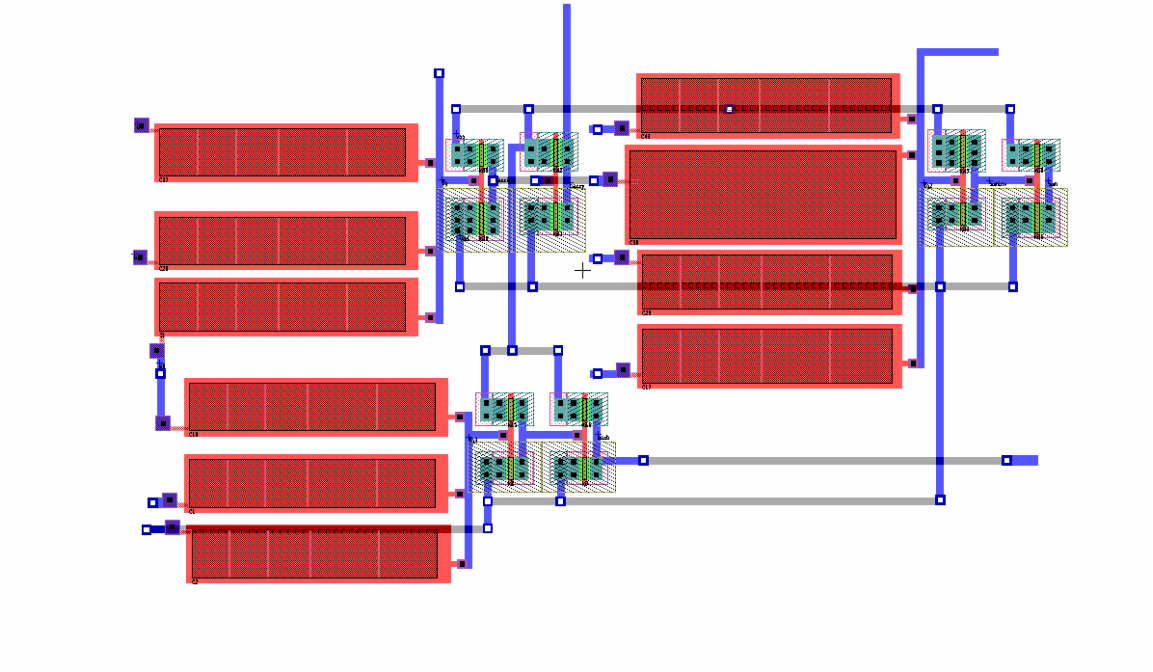


Figure 4.5: Layout of a full adder.

In our design, we have added buffers at the output of each stage of the 4-bit ALU, to improve delay of output signals. We have used buffers with the W_p/W_n increasing progressively with each stage [39]. At each stage of the 4-bit ALU, the output from the multiplexer stage at the output is taken to a buffer chain. Four inverters of increasing W_p/W_n ratios form the buffer chain. The W_p/W_n ratios are 9.6/7.6, 14.8/9.6, 16.4/13.2 and 20.8/13.2 where $L_p = L_n = 1.6\mu m$ The buffer stages also decrease distortion of the waveforms at the output stages.

In SPICE, simulations for the full adder circuit, the supply voltage V_{DD} is 3.0V and V_{SS} is 0.0V. Appendix A shows the SPICE LEVEL 3 MOS model parameters. The voltage levels of input pulse depicting '1' and '0' states are 3V and 0V, respectively. The input waveforms were pulse waveforms of following frequencies: 10MHz and 5MHz with 5 ns rise and fall times. The layout was extracted to obtain the SPICE netlist. Adding resistors as explained in Appendix C modified the netlist to solve the problem of DC convergence in SPICE. By adding appropriate input signals, transient analysis was performed. Figure 4.6 shows the waveforms of the input signals and the output waveforms of SUM and CARRY bits. Input A is a pulse of 10MHz. Input B a pulse of 5MHz. Input C_{IN} is a pulse of 5MHz. The SUM and CARRY bit waveforms are as shown in Fig. 4.6.

4.3 Arithmetic Logic Unit

The 4-bit ALU comprises of 4 to 1 and 2 to 1 multiplexers at the input and output sides and full adder with additional logic. The full adder is configured as ripple carry adder. S_0, S_1 and S_2 are the select signals that decide the operation being performed. The truth table for the eight operations performed by the ALU is shown in Table 4.3.

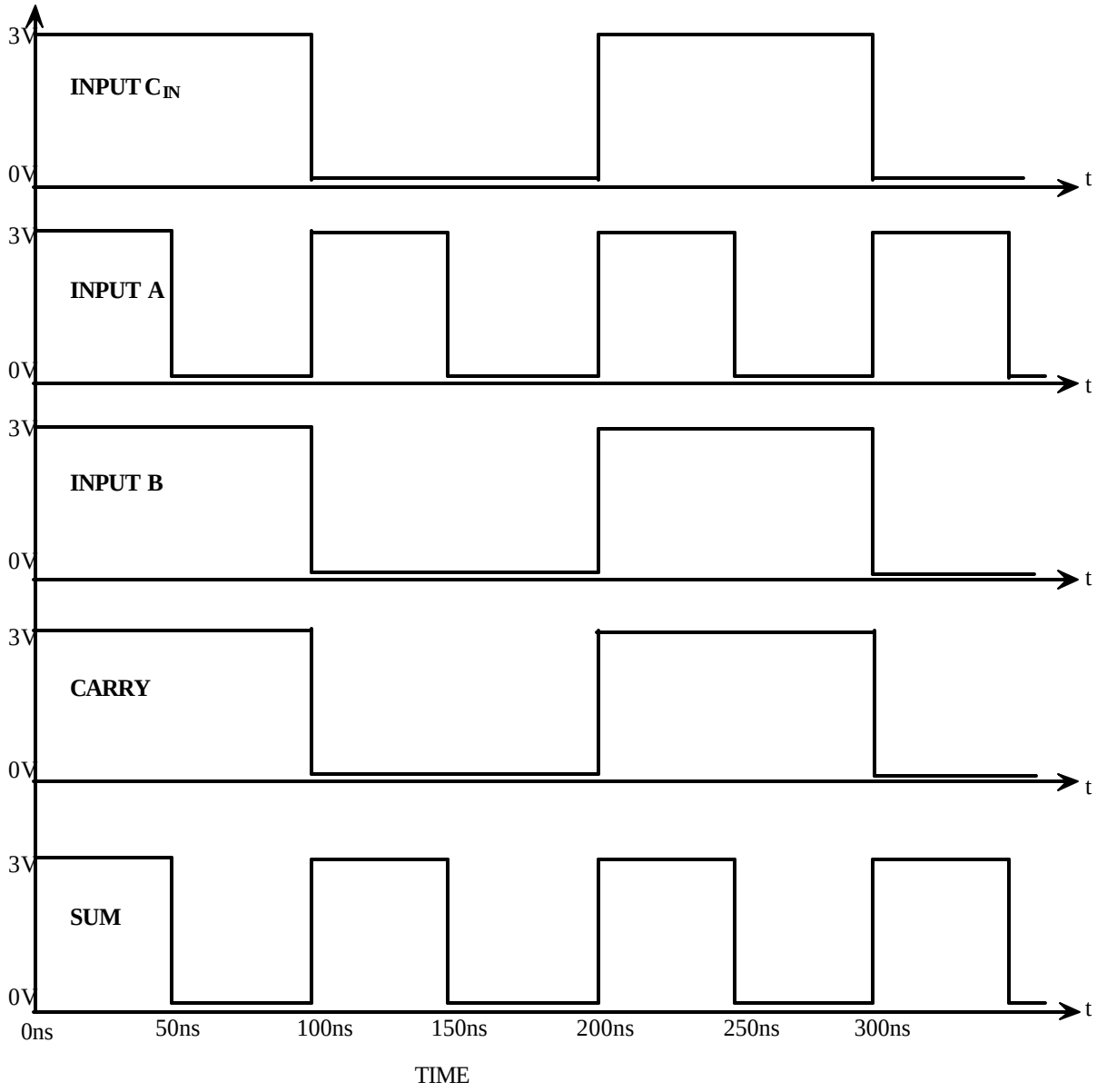


Figure 4.6: Post layout simulated waveforms for the full adder showing SUM and CARRY bits.

Table 4.3 Truth table of a 4-bit ALU

S_2	S_1	S_0	**Operation performed**
0	0	0	INCREMENT
0	0	1	DECREMENT
0	1	0	ADDITION
0	1	1	SUBTRATION
1	0	0	AND
1	0	1	OR
1	1	0	EXOR
1	1	1	EXNOR

Signal S_2 is connected to logic '0' for arithmetic operations, and connected to logic '1' for logical operations. The multiplexer logic at the output side of each stage of the ALU gives the final output. For logical operations the output of each stage is independent of the other stages. In case of arithmetic operations, the carry ripples from LSB to MSB position. Therefore the output of each stage depends on the previous stage.

For the layout of the 4-bit ALU, four stages of the full adder are cascaded in ripple carry adder configuration. The layout of the 4-bit ALU is shown in Fig. 4.7. The entire layout was placed in the 1.5µ padframe. Connections were made for inputs, outputs, supply voltages and ground pins on the padframe. The chip layout before fabrication is shown in Fig. 4.8. The design was fabricated in AMI 1.5µm CMOS process. The photograph of fabricated 4-Bit ALU chip is shown in Fig. 4.9. The 4-Bit ALU occupies approximately an area of 830 x 935 μm^2. The photograph of the chip including padframe is shown in Fig. 4.10.

SPICE simulations for the 4-bit ALU were done for post-layout extracted netlists. The post-layout extracted netlist was modified to overcome the DC convergence problem of the simulator. Two pulse trains of frequencies 10MHz and 5MHz were applied to the inputs. The select signal S_0 was tied to logic '0' for ARITHMETIC operations and tied to logic '1' for LOGIC operations. The post-layout SPICE simulation waveform for arithmetic operation ADDITION is shown in Fig. 4.11. For SUBTRACTION, two's complement method is used. The multiplexer stage at the input passes on the complement of input B to the full adder. The input, C_{IN} is set to '1'. Adding inputs A, B' (which is complement of B) and C_{IN}, in effect, performs a SUBTRACT operation.

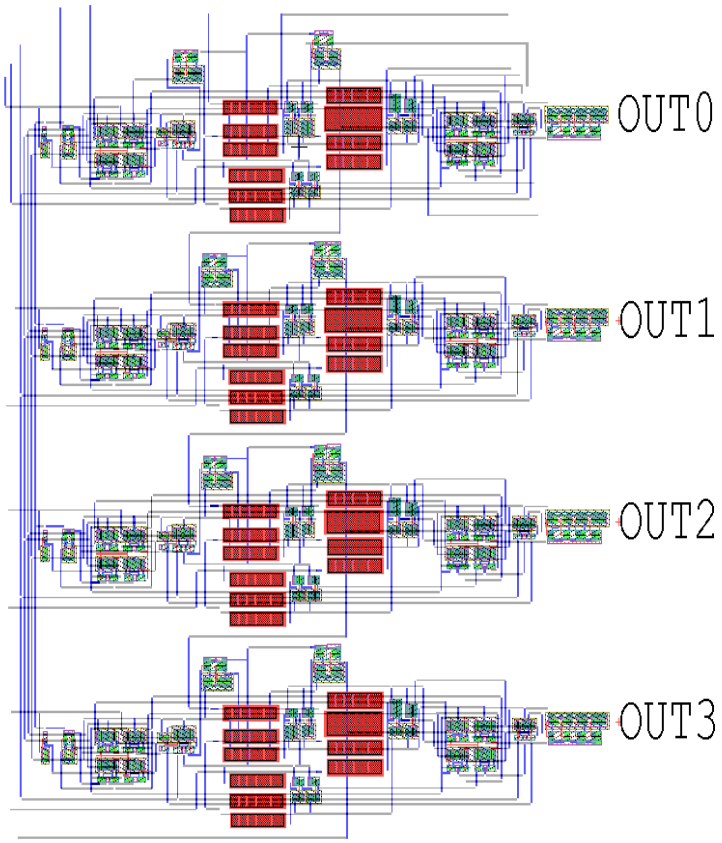
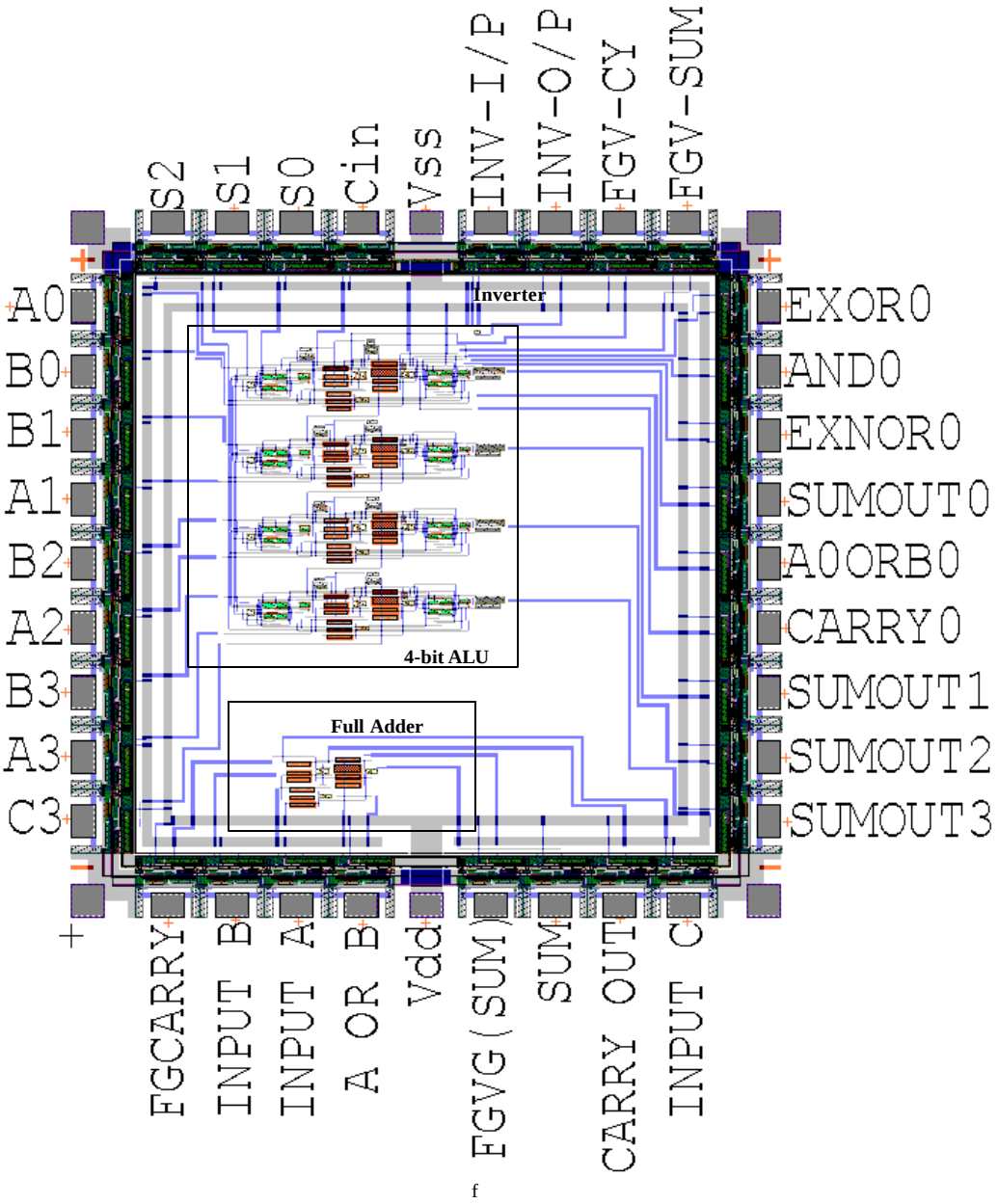


Figure 4.7: Layout of a 4-bit ALU.



f

Figure 4.8: Full chip layout for a 4-bit ALU.
Note: Individual building blocks such as inverter and FA are also included for testability.

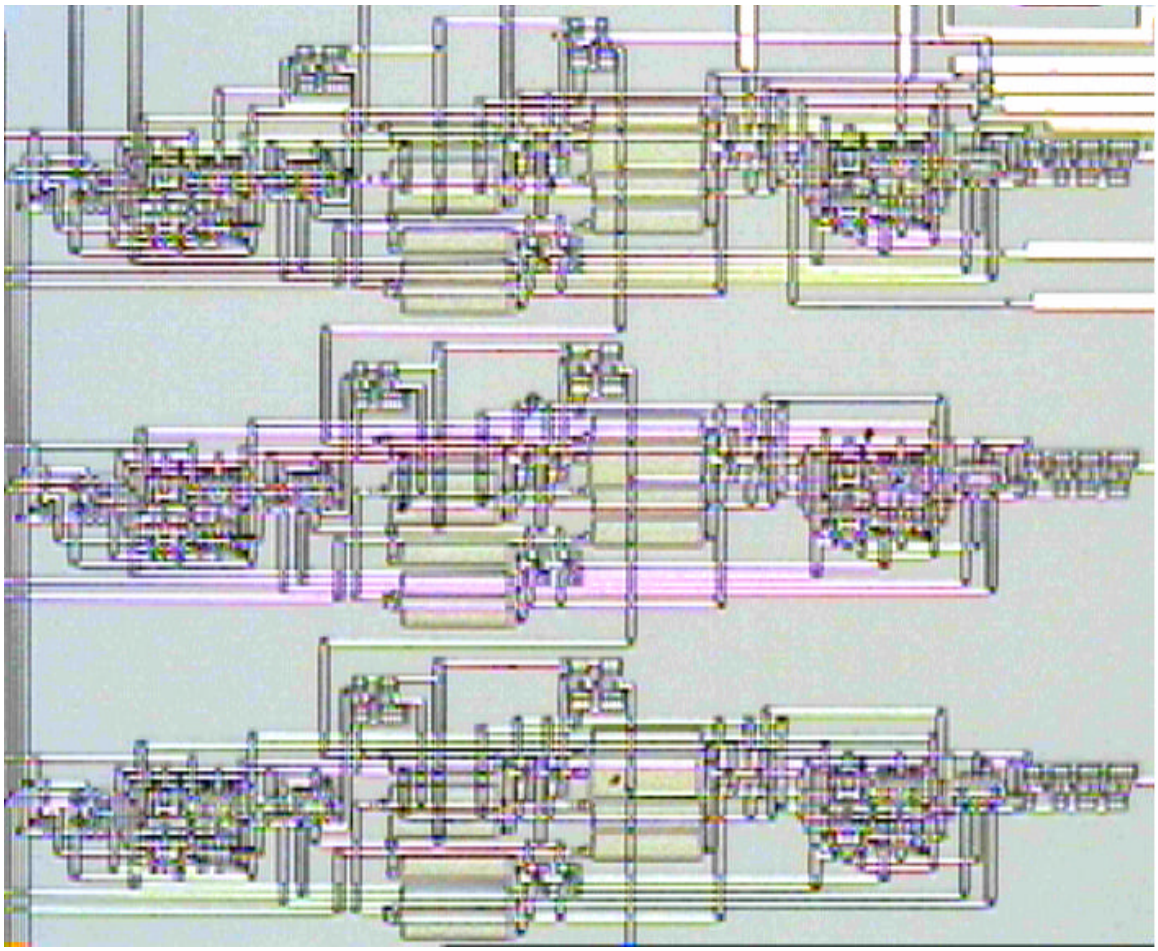


Figure 4.9: Photograph of a fabricated 4-Bit ALU chip.

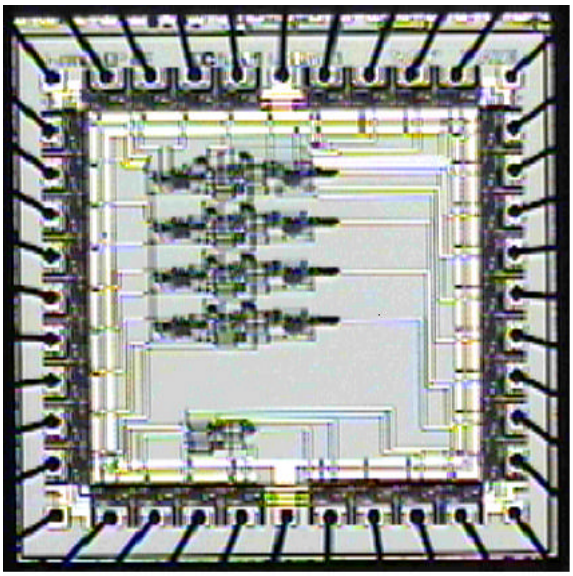


Figure 4.10: Chip photograph of a fabricated 4-Bit ALU and test devices.

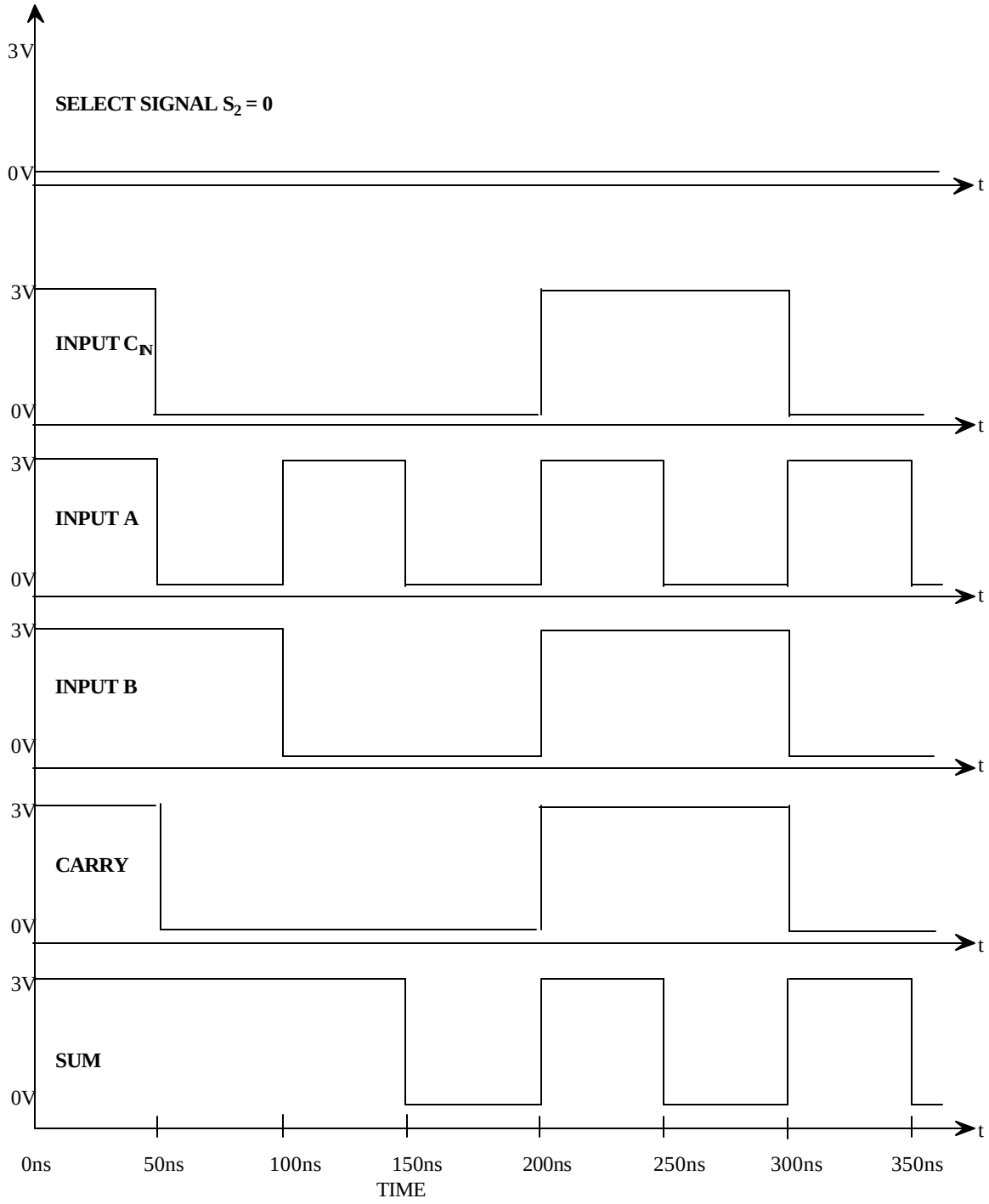


Figure 4.11: Post-layout waveforms for arithmetic operation (ADDITION).

Figure 4.12 shows the waveform for SUBTRACTION. Figure 4.13 shows the waveforms for DECREMENT operation. DECREMENT is also a SUBTRACTION operation with '1' being subtracted from operand A. Figure 4.14 shows the waveforms signals for INCREMENT operation. INCREMENT operation is addition of '1' to input A. The multiplexer passes a logic '1' to the full adder which will be added to input A thus incrementing it. For logical operations, the signal S_2 is connected to logic '1'. The input C_{IN} is set to logic '0'.

The full adder logic performs EXOR, AND, EXNOR and OR operations. Hence it gives four outputs. From the Boolean equations for SUM and CARRY bits, we realize that they are EXOR and AND operations respectively. The invert of EXOR would give us EXNOR. Additional logic as shown in Fig. 4.15 has been added to the full adder to realize OR operation. The multiplexer logic at the output stage passes on one of the four full adder outputs as the final output. Figure 4.16 shows the waveform for the four logical operations. The post-layout simulations were performed using model parameters of fabricated chip. The MOS model parameters of the fabricated chip are listed in Appendix B [35].

4.3.1 Delay Measurements for 4-Bit ALU

The 4-Bit ALU performs four arithmetic and four logical operations. Simulations were performed using SPICE simulator. All four stages of the 4-Bit ALU are identical. For logical operations, it is sufficient to measure the delay in only 1-bit stage of the ALU since each output bit is independent of the preceding stage. However for arithmetic operations, specific input patterns were chosen to obtain worst-case delay measurements.

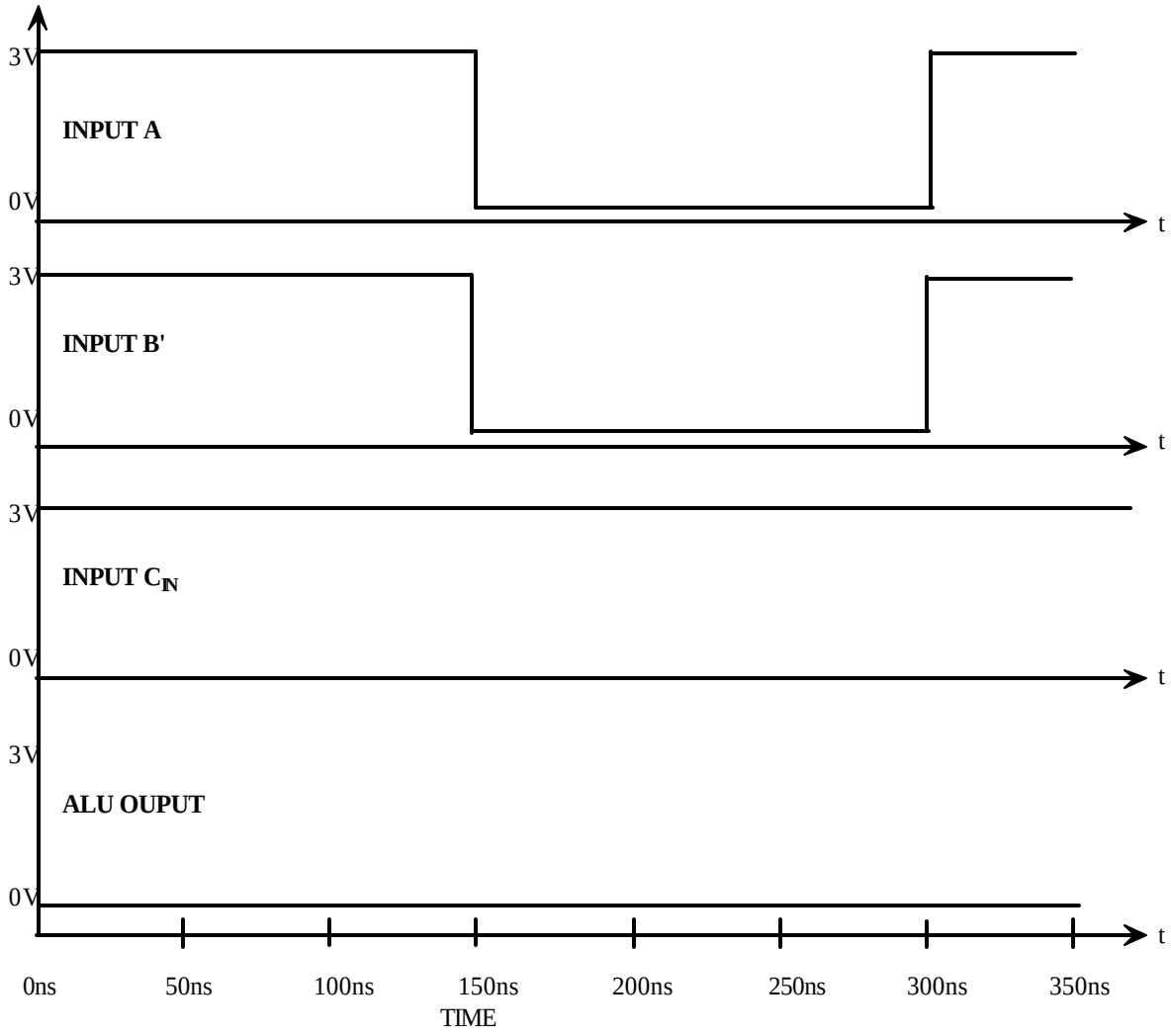


Figure 4.12: Post-layout waveforms for arithmetic operation (SUBTRACTION).

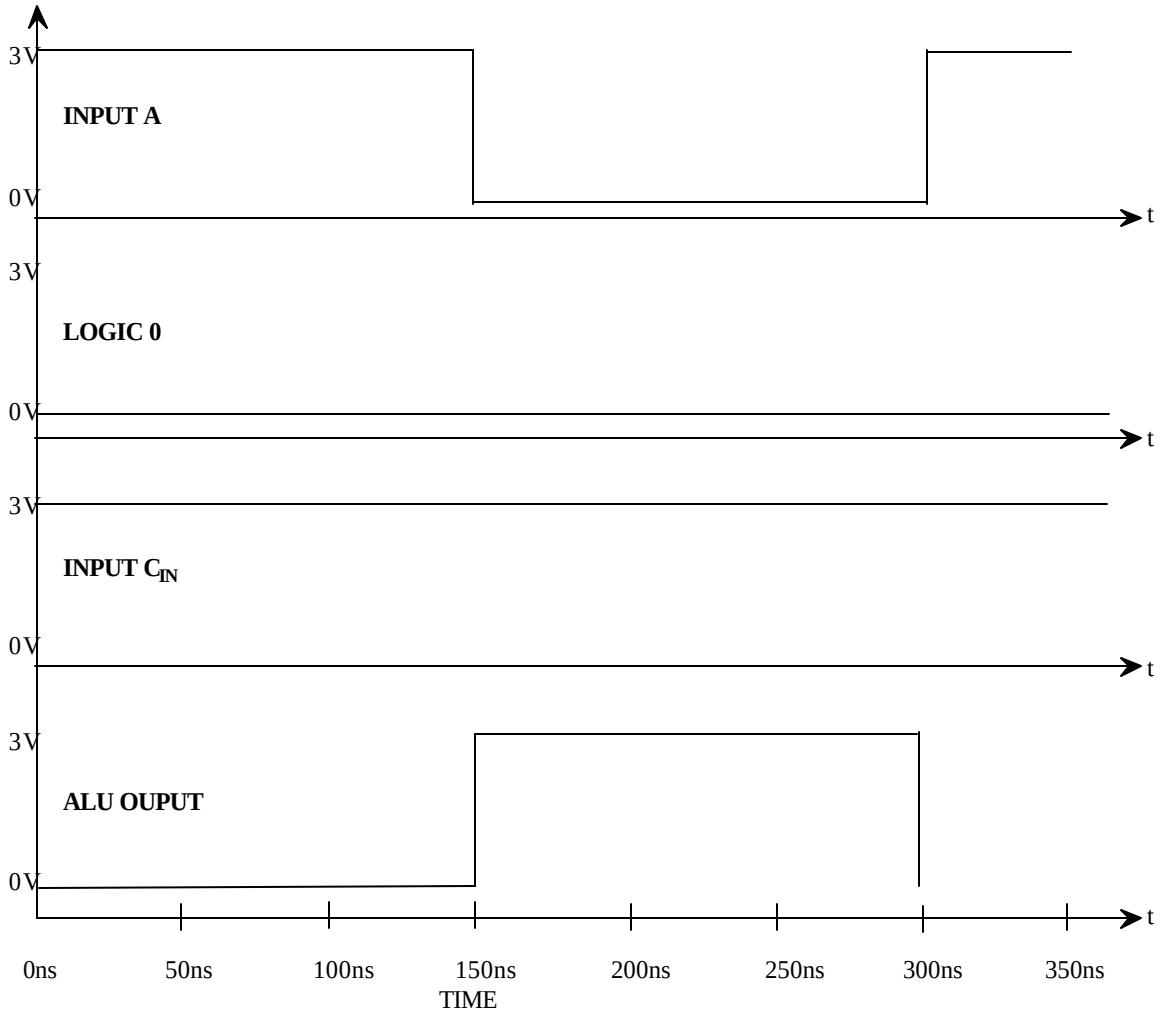


Figure 4.13: Post-layout waveforms for arithmetic operation (DECREMENT).

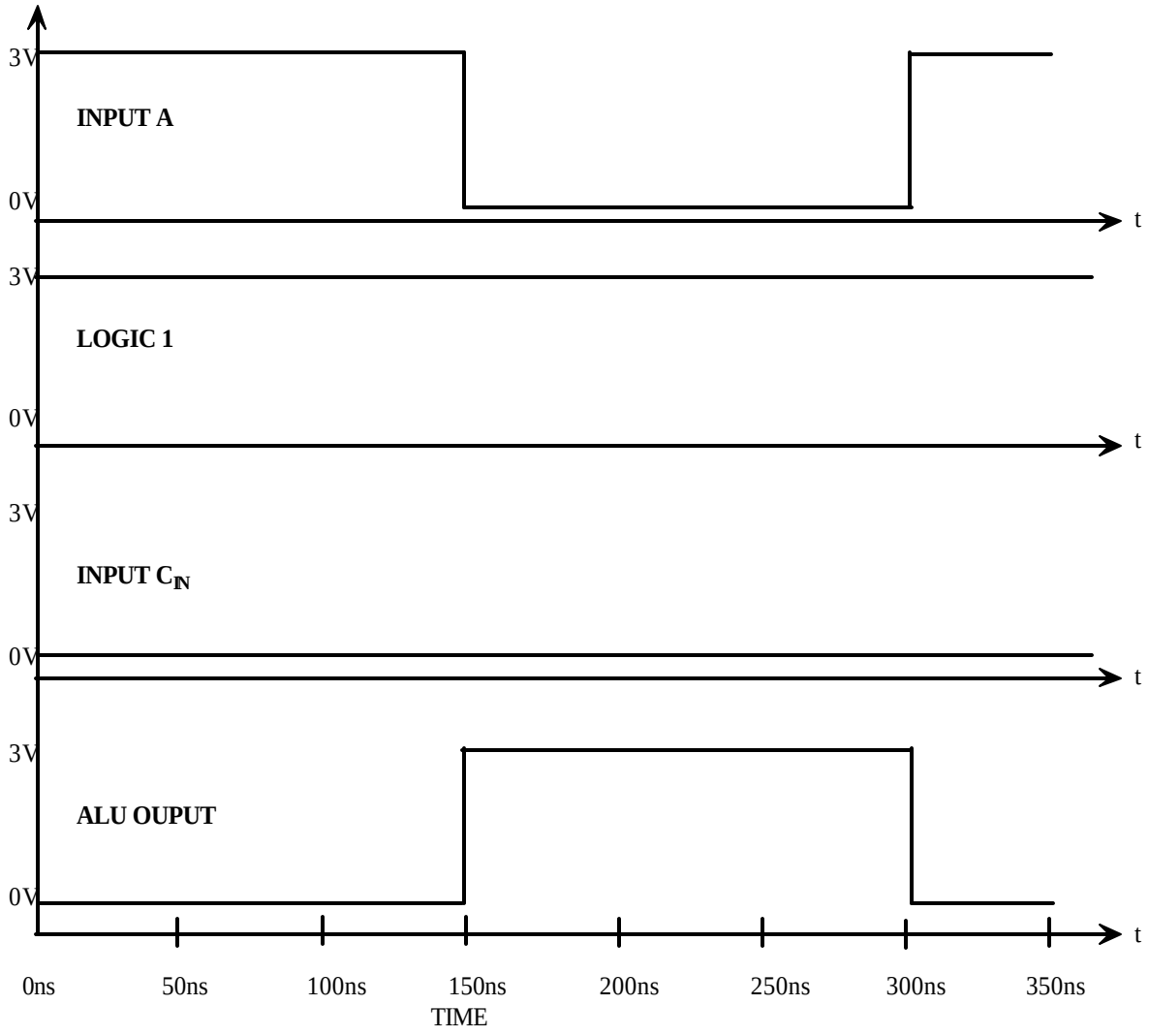
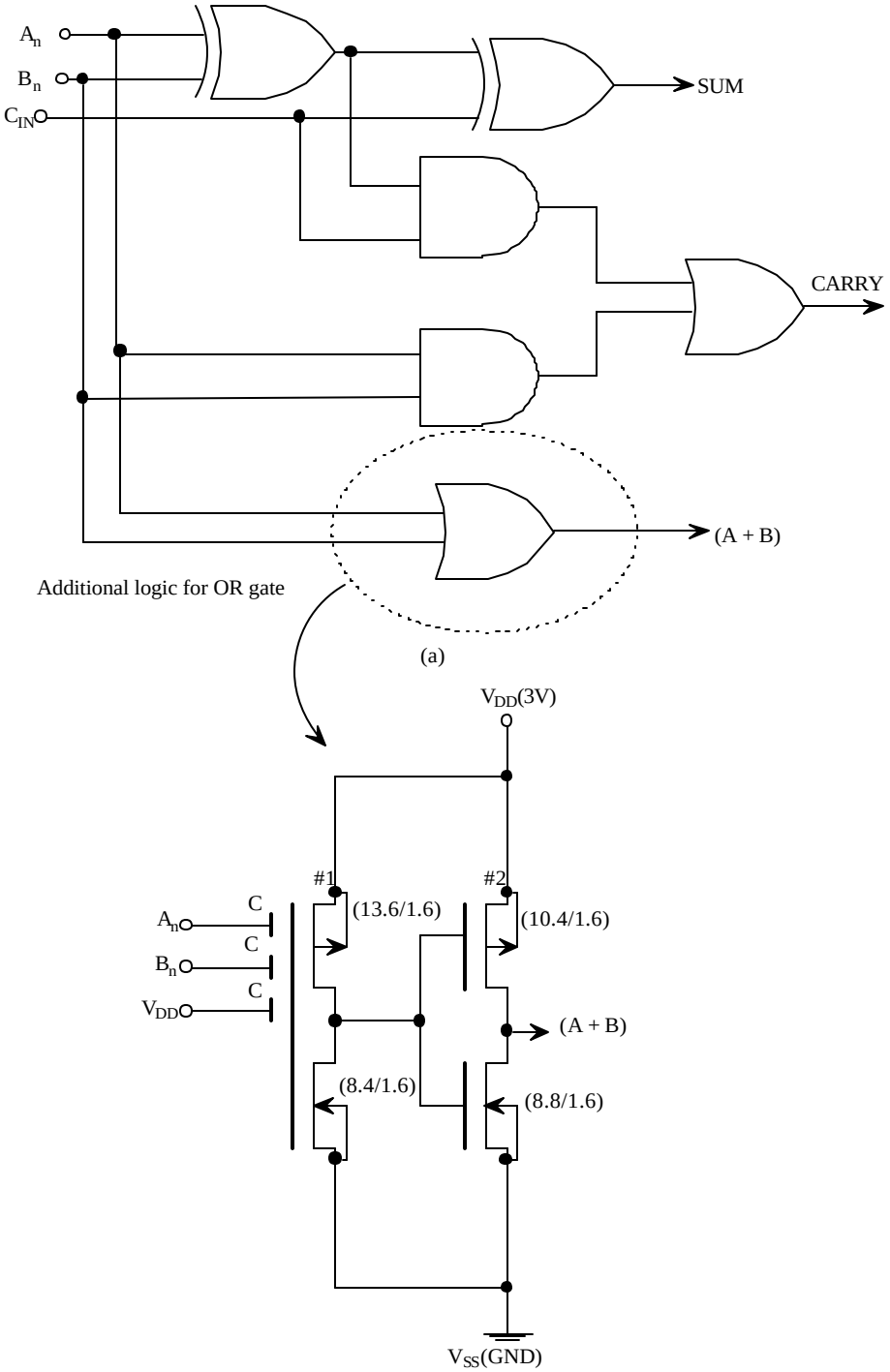


Figure 4.14: Post-layout waveforms for arithmetic operation (INCREMENT).



(b)

Figure 4.15: (a) Additional logic OR gate in full adder (b) MIFG CMOS equivalent circuit to realize OR operation.

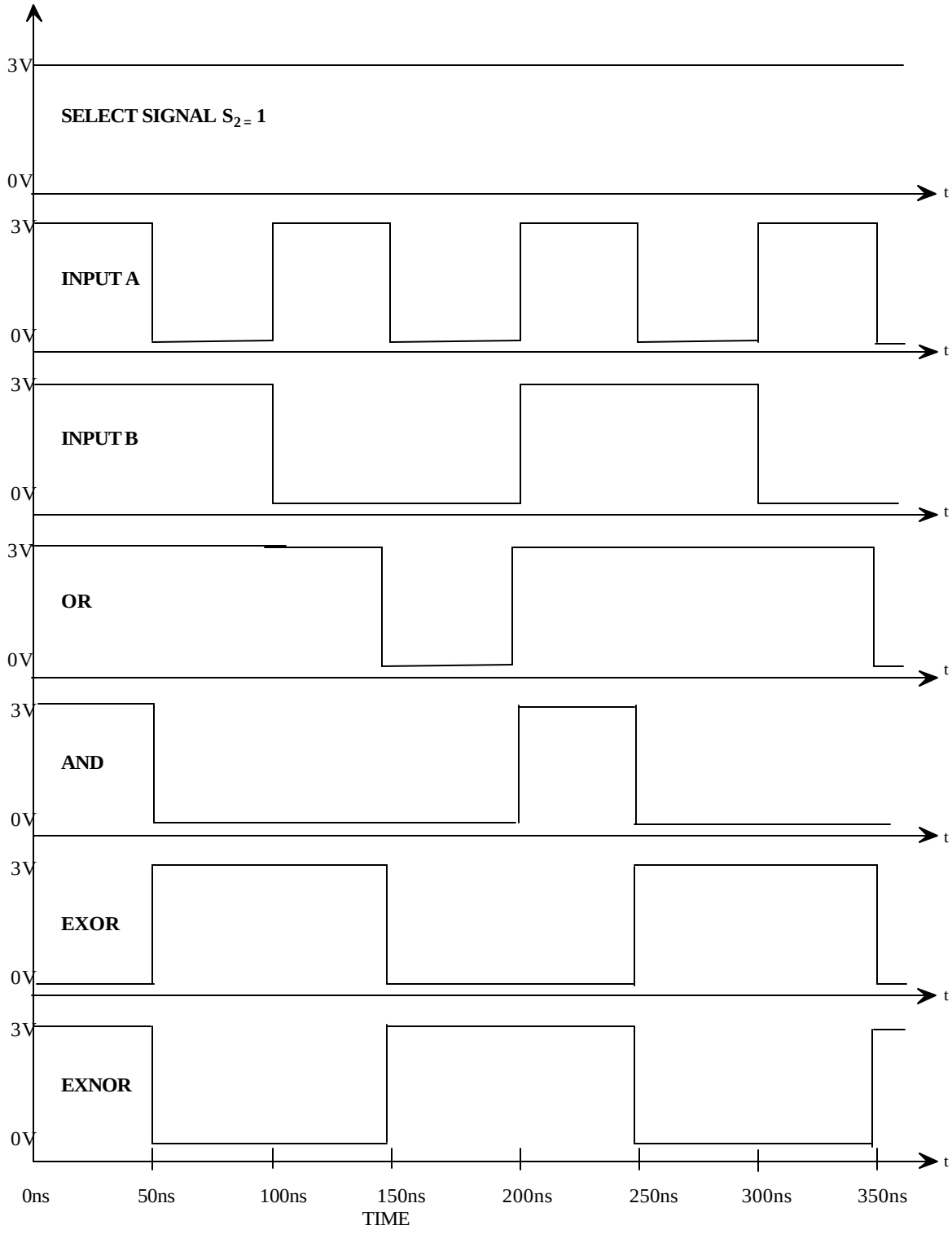


Figure 4.16: Post-layout waveforms for all LOGICAL operations.

Table 4.4 gives the input pattern for worst-case delays in arithmetic operations. In case of ADDITION, input patterns [A_3 A_2 A_1 A_0] = [0 0 01] and [B_3 B_2 B_1 B_0] = [1 1 1 1] and initial CARRY-IN as '0'. The carry propagates from LSB to MSB position. For INCREMENT operation the input pattern [A_3 A_2 A_1 A_0] = [1 1 1 1] when incremented by 1 generates a CARRY which trickles all the way from LSB to MSB. The worst-case delay input patterns for SUBTRACTION and DECREMENT are also chosen in the same way. Input pattern [A_3 A_2 A_1 A_0] = [1 0 0 0] and [B_3 B_2 B_1 B_0] = [1 1 1 1] generate the maximum delay. Since MSB of A (A_3) = logic '1' and LSB of B (B_0) = logic '0', the CARRY will have to propagate from MSB of A (A_3) to LSB of A (A_0) in order to perform SUBTRACT operation. Figure 4.17 shows the waveforms for SUM bit for post-layout simulation and experimental results. The delay between the two waveforms shows the worst-case delay. Figure 4.18 shows the waveforms for CARRY bit. Figure 4.19 shows the waveforms for logical operations EXOR and EXNOR. The delays presented are the delays for just a single stage of the 4-bit ALU. Logical operations do not have any worst-case delay, as each stage of the ALU is independent of the other. Figure 4.20 shows the post-layout and experimental waveforms for logical operations AND and OR. The delay in simulated waveforms has been calculated for a load capacitance of 15pF. The value 15pF is chosen because that is approximately how much capacitance the oscillator probe would provide in experimental setup. Table 4.5 has the delay measurements of arithmetic and logical operations tabulated for post-layout and experimental results

Table 4.4 Input patterns for worst-case delay measurements

	A_3 A_2 A_1 A_0 (LSB)	**B_3 B_2 B_1 B_0 (LSB)**
ADDITION	0 0 0 1	1 1 1 1
SUBTRACTION	1 0 0 0	1 1 1 1
DECREMENT	1 0 0 0	
INCREMENT	1 1 1 1	

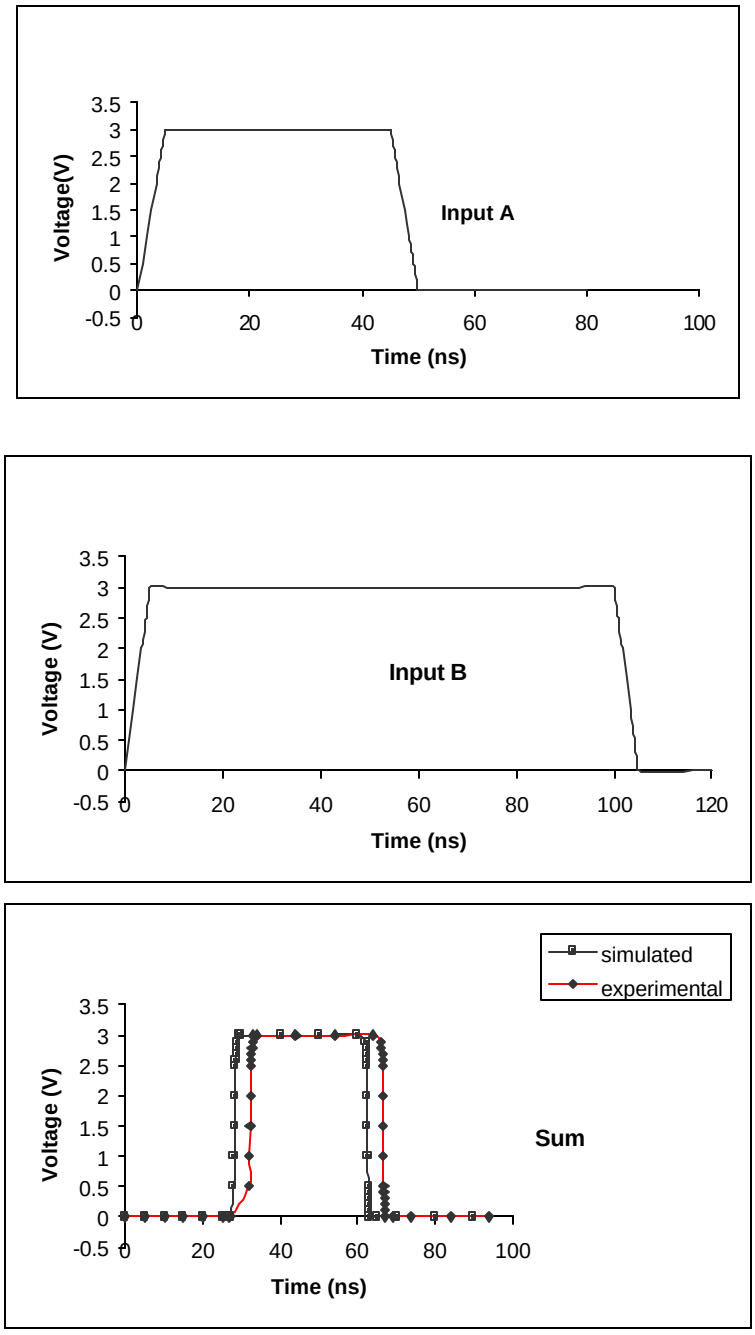


Figure 4.17: Waveforms for SUM bit showing worst-case delay between post-layout simulation and experimental results.

.

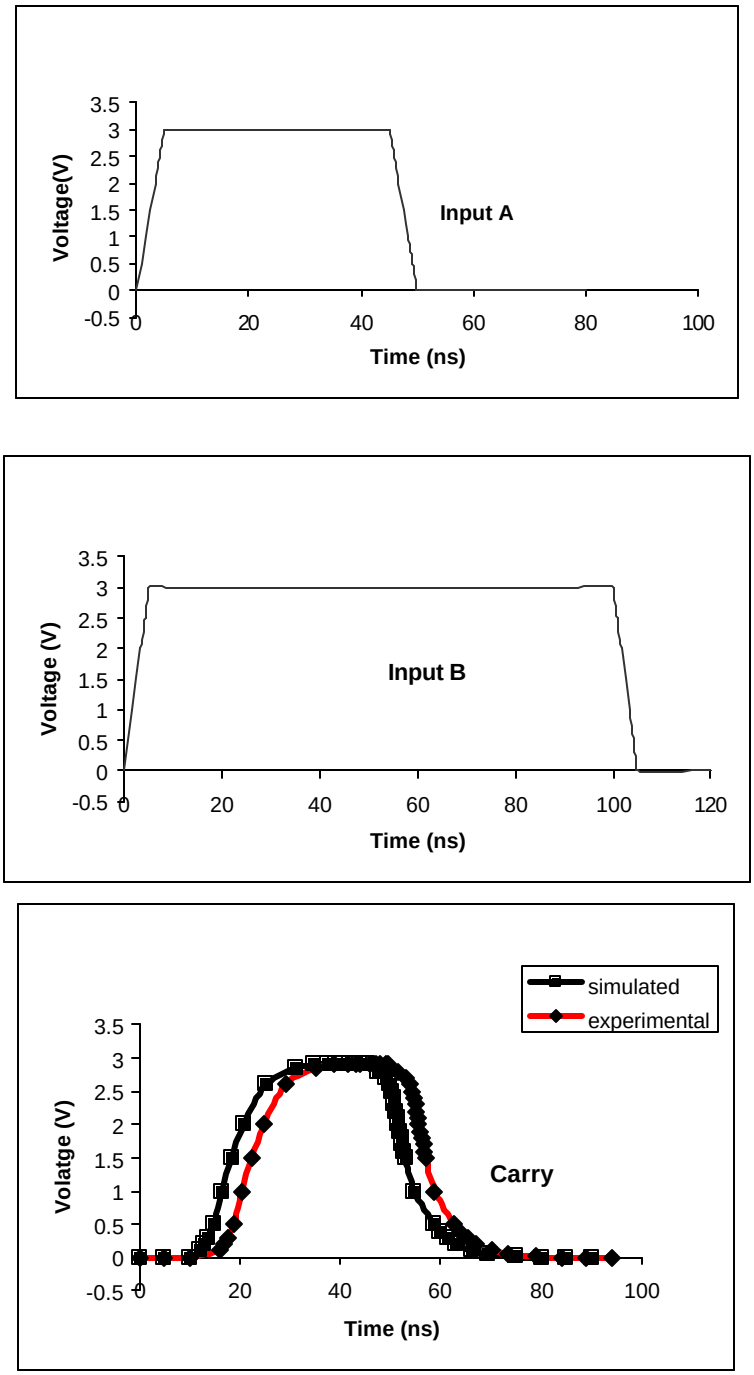


Figure 4.18: Waveforms for CARRY bit showing worst-case delay between post-layout simulation and experimental results.

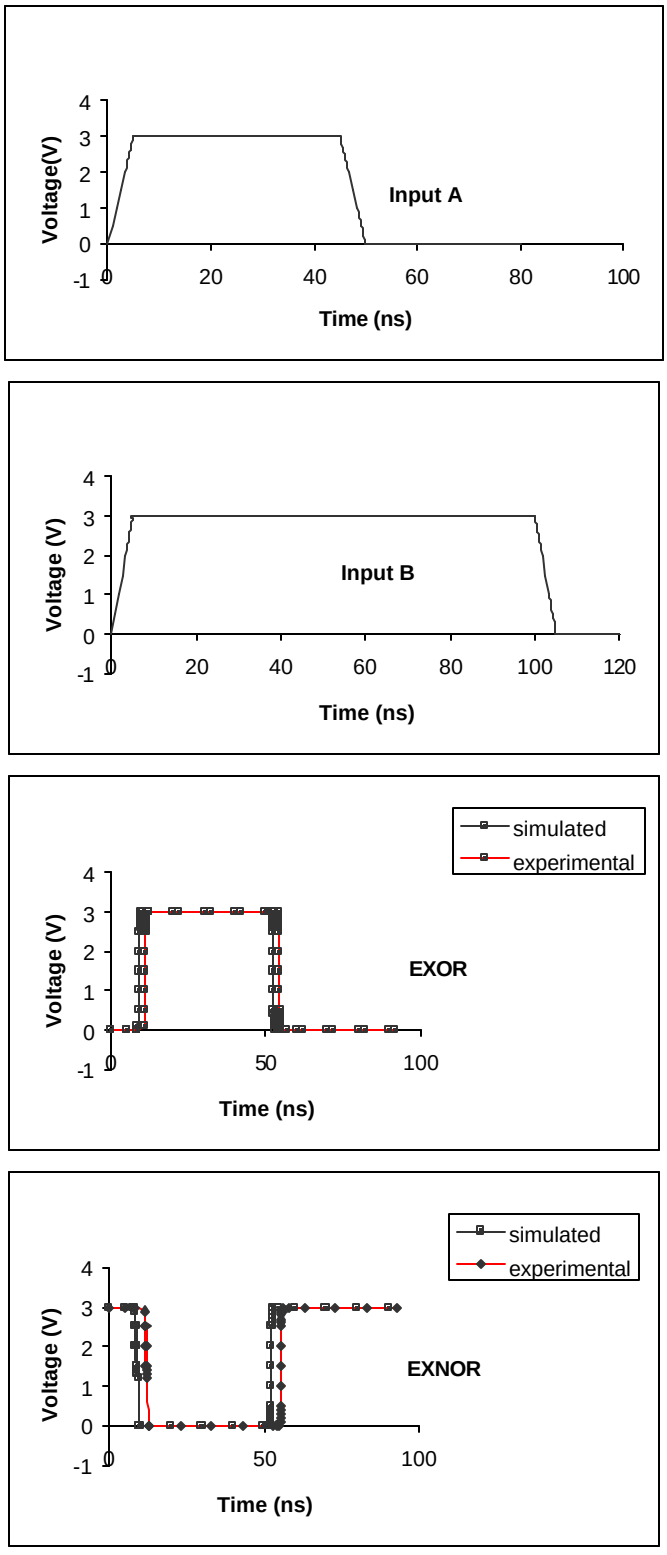


Figure 4.19: Waveforms for EXOR and EXNOR bits showing delay between post-layout simulation and experimental results.

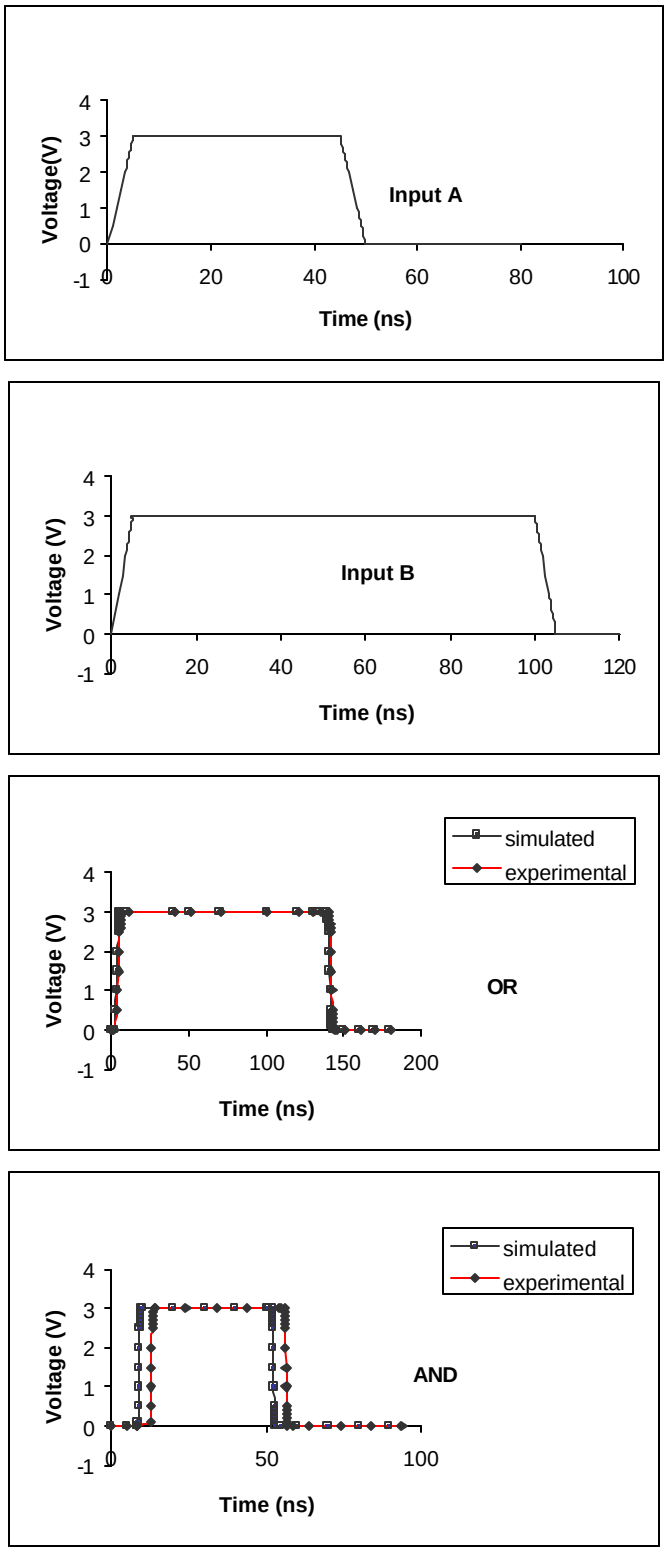


Figure 4.20: Waveforms for OR and AND bits showing delay between post-layout simulation and experimental results.

Table 4.5. Delay measurements for all output bits of ALU for arithmetic and logical operations for post-layout simulated and experimental results

OPERATION	**DELAY Tp (ns) SIMULATED with C_L = 15pf**	**DELAY Tp (ns) EXPERIMENTAL**
SUM	26	32
CARRY	17	22
EXOR	26	28
EXNOR	28	31
OR	27	28
AND	28	32

Chapter 5

Conclusion and Future Work

An integrated circuit design has been presented for a 4-bit arithmetic logic unit using multiple-input floating gate MOSFETs. The principle of MIFG transistors, calculating weighted sum of all inputs at gate level and switching transistors ON or OFF depending upon calculated floating gate voltage greater than or less than switching threshold voltage, is utilized.

The full integrated circuit is designed and simulated in standard 1.5µm digital CMOS technology. The circuits are simulated in SPICE with MOSIS LEVEL 3 MOS model parameters. The physical layout for all circuits is drawn using L-Edit version 8.03. The post-layout simulations included parasitic nodal capacitances to bring the simulation results as close as possible to the real time results.

The 4-bit ALU uses a ripple carry adder configuration. The full adder with additional logic for OR operation has been designed entirely using multi-input floating gate transistors. Compared with earlier reduced transistor designs for full adder, our full adder design implemented with multi-input floating gate transistors, uses at least 43% lesser transistors. When compared to conventional designs the reduction in number of transistors is nearly 84%. Table 5.1 shows a comparison in transistor count between earlier designs and our design. With 15pF simulated capacitive load, the maximum propagation delay is 32ns for the SUM bit. The power dissipation is close to 0.5mW.

5.1 Future work

An important aspect of designing circuits using floating gate devices is determining the value of unit capacitance ‘C’, which affects the layout area and

Table 5.1 Comparison of between earlier full adder designs and our design with respect to transistor count

Full adder Design	**Number of transistors used**
Bui *et al* ‘s design [9]	10
Wang *et al*’s design [10]	12
Shams *et al* ‘s design [11]	16
Radhakrishna’s design [12]	12
MIFG full adder design [present]	8

performance of the circuits. Future work could be extended to optimizing unit capacitance value. Knowing the fabrication run being used and error percentage in process would give us feedback for designing with smaller capacitors. Realizing logic designs with smaller capacitors will reduce the power dissipation and make the layout more area efficient. Further work could also include extending the current design to include more number of input bits for the ALU. The practical design aspect of simulating floating gate MOSFET in SPICE using low-level models is still an issue. Manufactures do not provide models for simulating floating gate MOSFETs, hence a special technique to simulate these devices with standard MOS models is required.

Bibliography

[1] W. Wolf, *Modern VLSI Design- Systems On Silicon*, Prentice Hall, 1998.

[2] Y. Tsividis, *Operation and Modeling of The MOS Transistor*, Mc Graw-Hill, 1999.

[3] T. Shibata and T. Ohmi, "A functional MOS transistor featuring gate-level weighted sum and threshold operations", *IEEE Trans. on Electron Devices*, vol. 39, Issue 6, pp.1444-1455, June 1992.

[4] A. Srivastava, H.N. Venkata and P.K. Ajmera, "A novel scheme for a higher bandwidth sensor readout", *Proc. of SPIE*, vol. 4700, pp. 17-28, 2002.

[5] W. Weber, S.J. Prange, R. Thewes, E. Wohlrab and A. Luck, "On the application of the neuron MOS transistor principle for modern VLSI design", *IEEE Trans. on Electron Devices*, vol. 43, pp.1700 – 1708, Oct. 1996.

[6] L. Yin, S.H.K. Embadi and E. Sanchez-Sinencio, "A floating gate MOSFET D/A converter", *Proc. of IEEE International symposium on Circuits and Systems*, vol. 1, pp. 409-412, June 1997.

[7] E. Rodriguez-Villegas, G. Huertas, M.J. Avedillo, J. M. Quintana and A. Rueda, "A practical floating-gate Muller-C element using vMOS threshold gates", *IEEE Trans. on Circuits and Systems-II: Analog and Digital Signal Processing*, vol. 48, pp. 102-106, Jan. 2001.

[8] J.M. Rabaey, *Digital Integrated Circuits- A Design Perspective*, Prentice Hall, 1996.

[9] H.T. Bui, Y. Wang and Y. Jiang, "Design and analysis of low-power 10-transistor full adders using novel XOR-XNOR gates", *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 49, pp. 25 –30, Jan. 2002.

[10] J.M. Wang, S.C. Fang and W.C. Fang, "New efficient designs for XOR and XNOR functions on transistor level", *IEEE J. of Solid State Circuits*, vol. 29, pp. 780-786, July 1994.

[11] A.M. Shams and M.A. Bayoumi, "A novel high-performance CMOS 1-bit full-adder cell", *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 47, pp. 478 –481, May 2000.

[12] D. Radhakrishnan, "Low-voltage low-power CMOS full adder", *IEE Proceedings-Circuits, Devices and Systems*, vol.148, pp. 19 -24, Feb. 2001.

[13] K. Suzuki, M. Yamashina, J. Goto, Y. Inoue, T. Koseki, Y. Horiuchi, T. Hamatake, K. Kumagai, T. Enomoto and H. Yamada, "A 2.4-ns, 16-bit, 0.5-ìm CMOS arithmetic logic unit for microprogrammable video signal processor LSIs", *Proc. of the IEEE Custom Integrated Circuits Conference*, vol. 9, pp.12.4.1 -12.4.4, May 1993.

[14] A. Srivastava and C. Srinivasan, " ALU Design using Reconfigurable CMOS Logic", *Proc. of the 45th IEEE 2002 Midwest Symposium on Circuits and Systems*, vol.2, pp. 663-666, Aug. 2002.

[15] R.R. Harrison, J.A. Bragg, P. Hasler, B.A. Minch and S.P. Deweerth, "A CMOS programmable analog memory-cell array using floating-gate circuits" *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 48, pp. 4 –11, Jan. 2001.

[16] Y. Berg, T.S. Lande and O. Naess, "Programming floating-gate circuits with UV-activated conductances", *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 48, no. 1, pp. 12 –19, Jan. 2001.

[17] S.A. Jackson, J.C. Killens and B.J. Blalock, "A Programmable current mirror for analog trimming using single poly floating-gate devices in standard CMOS technology", *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 48, no. 1, pp. 100 –102, Jan. 2001.

[18] B.A. Minch, P. Hasler and C. Dioro, "Multiple-input translinear element networks**",** *Proc. of IEEE International Symposium on Circuits and Systems*, (ISCAS '98), vol. 1, pp. I 88 –91, June 1998.

[19] Y.Berg, O. Naess and M. Hovin, "Ultra low-voltage floating-gate transconductance amplifier", *IEEE Trans. on Circuits and Systems-II: Analog and Digital Signal Processing*, vol. 48, no. 1, pp. 37-44, Jan. 2001.

[20] P. Hasler, "Continuous-time feedback in floating-gate MOS circuits", *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 48, no. 1, pp. 56 –64, Jan. 2001.

[21] P. Hasler, B.A. Minch and C.Diorio, "An autozeroing floating-gate amplifier", *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 48, no. 1, pp. 74 –82, Jan. 2001.

[22] T. Shibata and T. Ohmi, "Neuron MOS. binary-logic integrated circuits-Part I: Design fundamentals and soft-hardware logic circuit implementation", *IEEE Trans. on Electron devices*, vol. 40, no. 3, pp. 570-575, March 1993.

[23] T. Shibata and T. Ohmi, "Neuron MOS. binary-logic integrated circuits-Part II: "Simplifying techniques of circuit configuration and their practical applications", *IEEE Trans. on Electron devices*, vol. 40, no. 5, pp. 974-979, May 1993.

[24] H.R. Mehrvarz and C.Y. Kwok, "A novel multi-input floating-gate MOS four-quadrant analog multiplier", *IEEE J. of Solid-State Circuits*, vol. 31, Issue 8, pp.1123-1131, Aug. 1996.

[25] L.S.Y. Wong, C.Y. Kwok and G.A. Rigby, "A 1V CMOS D/A Converter with multi-input floating-gate MOSFET", *IEEE J. of Solid-State Circuits*, vol.34, no. 10, pp. 1386-1390, Oct. 1999.

[26] K. Ogawa, T. Shibata, T. Ohmi, M. Takatsu and N. Yokoyama, "Multiple-input neuron MOS operational amplifier for voltage-mode multi-valued full adders", *IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing*", vol. 45, no. 9, pp. 1307 –1311, Sep. 1998.

[27] A. Luck, S. Jung, R. Brederlow, R.Thewes, K. Goser and W. Weber, " On the design robustness of threshold logic gates using multi-input floating gate MOS transistors", *IEEE Trans. on Electron Devices*, vol. 47, pp. 1231-1239, June 2000.

[28] B.A. Minch, "A folded floating-gate differential pair for low-voltage applications**",** *Proc. of the IEEE International Symposium on Circuits and Systems* (ISCAS 2000), vol. 4, pp. IV 253-256, May 28-31, 2000.

[29] P. Hasler, B.A. Minch and C. Diorio, "Floating-gate devices: they are not just for digital memories any more", *Proc. of the IEEE International Symposium on Circuits and Systems*, pp. II 388–391, July 1999.

[30] T.Shibata and T.Ohmi, "An intelligent MOS. transistor featuring gate-level weighted sum and threshold operations", *IEDM Tech. Dig*, pp. 919-922, 1991.

[31] H.N. Venkata, "Ternary and quaternary logic to binary bit conversion CMOS integrated circuit design using multiple input floating gate MOSFETs", M.S (EE) Thesis, ECE Department, Louisiana State University, Baton Rouge, Dec. 2002.

[32] A. Rantala, S. Franssila, K. Kaski, J. Lampinen, M. Aberg and P. Kuivalainen "Improved neuron MOS-transistor structures for integrated neural network circuits", *IEE Proceedings- Circuits, Devices and Systems*, vol. 148, pp 25-34, Feb. 2001.

[33] N. H. E. Weste and K. Eshranghian, *Principles of CMOS VLSI Design- A Systems Perspective*, Addison-Wesley, 1998.

[34] Y. Tsividis, *Mixed Signal Analog-Digital VLSI Devices and Technology: An Introduction*, Mc-Graw Hill, 1996.

[35] http://www.mosis.org.

[36] J. Ramirez-Angulo, G. Gonzalez-Altamirano and S.C. Choi, "Modeling multiple input floating gate transistors for analog signal processing", *Proc. of IEEE International Symposium on Circuits and Systems*, pp. 2020-2023, June 9-12, 1997.

[37] A. Srivastava and D. Govindarajan, "A fast ALU design in CMOS for low voltage operation", *J. of VLSI Design*, vol. 14, no. 4, pp. 315-327, 2002.

[38] M. Morris Mano, *Digital Logic and Computer Design*, Prentice Hall, 1979.

[39] R. J. Baker, H.W. Li and D.E. Boyce, *CMOS Circuit Design, Layout and Simulation*, IEEE Press, 1998.

Appendix A

MOSFET Model Parameters [35]

nMOS Model Parameters

.MODEL NMOS NMOS (LEVEL = 3

+ TOX = 3.05E-8	NSUB = 1E15	GAMMA = 0.7292975
+ PHI = 0.7	VTO = 0.6052099	DELTA = 0.4903542
+ UO = 800	ETA = 9.999921E-4	THETA = 0.0710077
+ KP = 7.401542E-5	VMAX = 3.503416E5	KAPPA = 1.0708
+ RSH = 0.0299117	NFS = 4.770051E11	TPG = 1
+ XJ = 3E-7	LD = 7.15092E-12	WD = 6.615428E-7
+ CGDO = 1.76E-10	CGSO = 1.76E-10	CGBO = 1E-10
+ CJ = 2.821829E-4	PB = 0.8883154	MJ = 0.5
+ CJSW = 1.244664E-10	MJSW = 0.05)	

pMOS Model Parameters

.MODEL PMOS PMOS (LEVEL = 3

+ TOX = 3.05E-8	NSUB = 5.327947E16	GAMMA = 0.4898253
+ PHI = 0.7	VTO = -0.8651299	DELTA = 0.3307574
+ UO = 159.6662211	ETA = 1	THETA = 0.1329612
+ KP = 2.470055E-5	VMAX = 3.702724E5	KAPPA = 10
+ RSH = 27.1878946	NFS = 5.440123E11	TPG = -1
+ XJ = 2E-7	LD = 1E-12	WD = 9.994094E-7
+ CGDO = 2.25E-10	CGSO = 2.25E-10	CGBO = 1E-10
+ CJ = 3.022339E-4	PB = 0.7508359	MJ = 0.4334195
+ CJSW = 1.781647E-10	MJSW = 0.1286211)	

Appendix B

MOSFET Model Parameters of the Fabricated Chip [35]

nMOS Model Parameters

. MODEL CMOSN NMOS(	LEVEL = 3	
+ TOX = 3.07E-8	NSUB = 2.75325E15	GAMMA = .7620845
+ PHI = 0.7	VTO = 0.6298903	DELTA = 0.8569392
+ UO = 702.9336344	ETA = 9.99916E-4	THETA = 0.0734963
+ KP = 7.195017E-5	VMAX = 2.766785E5	KAPPA = 0.5
+ RSH = 0.0474566	NFS = 6.567094E11	TPG = 1
+ XJ = 3E-7	LD = 4.271014E-12	WD = 7.34313E-7
+ CGDO = 1.75E-10	CGSO = 1.75E-10	CGBO = 1E-10
+ CJ = 2.944613E-4	PB = 0.9048351	MJ = 0.5
+ CJSW = 1.236957E-10	MJSW = 0.05)	

pMOS Model Parameters

. MODEL CMOSP PMOS (	LEVEL = 3	
+ TOX = 3.07E-8	NSUB = 1E17	GAMMA = .4940829
+ PHI = 0.7	VTO = -0.8615406	DELTA = 0.5236605
+ UO = 250	ETA = 7.55184E-3	THETA = 0.1344949
+ KP = 2.438731E-5	VMAX = 9.345228E5	KAPPA = 200
+ RSH = 36.5040447	NFS = 5.518964E11	TPG = -1
+ XJ = 2E-7	LD = 9.684773E-12	WD = 1E-6
+ CGDO = 2.09E-10	CGSO = 2.09E-10	CGBO = 1E-10
+ CJ = 2.965467E-4	PB = 0.744678	MJ = 0.4276703
+ CJSW = 1.619193E-10	MJSW = 0.1055522)	

Appendix C

Simulating Floating Gate MOS Device

One of the practical design issues concerning circuits that implement floating gate devices is electrical simulation. Manufacturers of the simulation tools do not provide models for floating gate devices. This requires from us to devise methods of simulating floating gate device circuits. The difficulty in simulating MIFG devices is the inability of the simulator to converge when floating node exists. Most circuit simulators replace the input coupling capacitors in floating gate devices with open circuits during DC analysis. Simulation ends at this point, as it is not able to converge. Solution to the problem of floating gate MOSFET is presented in several paper providing solutions for this problem [6,7,36]. Rodriguez-Villegas *et al* [7] have introduced an initial operating point in floating gate circuits by using additional network formed by resistors and voltage-controlled voltage sources (VCVS). A transient analysis is performed which starts of with all inputs and circuit power supplies set to 0V. This initializes the voltage on the floating gate to zero. Later in the transient analysis, a stationary state is reached and this is used as fixed operation point for the rest of the simulations. Ramirez-Angulo *et al* [36] have used a very high resistor element between the floating gate and ground along with voltage elements in the circuit to overcome this problem. Yin *et al* [6] have shown a very simple simulation model for the floating gate devices. Using the model in [6], we have simulated a 4-input floating gate CMOS inverter circuit of Fig. C.1. In Fig. C.1, V_1, V_2, V_3 and V_4 are the voltages of the input gates of the MIFG CMOS inverter. V_F is the voltage on the floating gate. C_1, C_2, C_3 and C_4 are the input capacitors of 500fF

each. To avoid the DC convergence problem, resistor element, R_1 to R_4 are inserted across each of the four input capacitors. For this model, the relationship between the floating gate voltage and input gate voltages under the DC condition is given by equation C.1.

$$G_1(V_1 - V_F) + G_2(V_2 - V_F) + G_3(V_3 - V_F) + G_4(V_4 - V_F) = 0 \qquad \text{(C.1)}$$

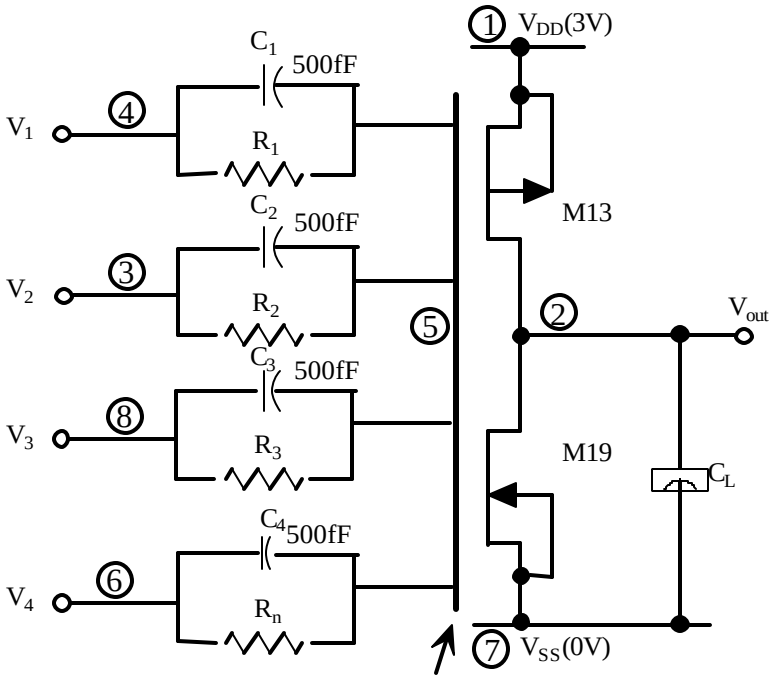
where $G_i = 1/R_i = k_i C_i$ and $G_{TOT} = \sum_{i=0}^{n} G_i$

where k_i is a constant.

Equation (C.1) reduces to the following form,

$$V_F = \frac{V_1 \times C_1 + V_2 \times C_2 + V_3 \times C_3 + V_4 \times C_4}{C_1 + C_2 + C_3 + C_4} \qquad \text{(C.2)}$$

In the circuit of Fig. C.1, R_i is selected as $R_i = 1/ k_i C_i$. The constant, k_i is chosen to make R_i as high as possible. In our design to make R_i in the range of 1000G, k is chosen as 10^{12}. The SPICE netlist for the circuit of Fig. C.1 is described as follows. Figure C.2 shows the input and output waveforms generated by simulating the circuit of Fig. C.1.



$V_{F:}$ Voltage on floating gate

Figure C.1: Simulation model of a multi-input floating gate CMOS inverter.

SPICE netlist for a 4-input floating gate CMOS inverter of the circuit of Fig. C.1

```
* Circuit Extracted by Tanner Research's L-Edit V8.20 / Extract V8.20 ;
* TDB File:  Z:\grad\chandra\1.5layout\alulayoutprefab.tdb
* Cell:  fgmosfet1       Version 1.5
* Extract Definition File:  C:\Program Files\Tanner EDA\L-Edit
Pro\tech\mosis\mAMI150.ext
* Extract Date and Time:  04/15/2002 - 20:16
* NODE NAME ALIASES
*       1 = vdd (-5,-7)
*       2 = Vout (22,35)
*       3 = V2 (-257,20)
*       4 = V1 (-252,81)
*       5 = fg (-19,37)
*       6 = V4 (-255,-108)
*       7 = vss (-8,73)
*       8 = V3 (-254,-44)
Cpar1 1 0 5.97664f
Cpar2 2 0 5.76992f
Cpar3 3 0 1.53216f
Cpar4 4 0 1.53216f
Cpar5 5 0 152.80944f
Cpar6 6 0 1.53216f
Cpar7 7 0 4.35936f
Cpar8 8 0 1.53216f

r1 4 5 1e12ohms
r2 3 5 1e12ohms
r3 8 5 1e12ohms
r4 6 5 1e12ohms
r5 0 5 1e12ohms

Vdd 1 0 dc 3v
Vss 7 0 dc 0v

V1 4 0 pulse(3V 0V 0N 1N 1N 60N 120N)
V2 3 0 pulse(0V 3V 0N 1N 1N 60N 120N)
V3 8 0 pulse(3V 0V 0N 1N 1N 60N 120N)
V4 6 0 pulse(3V 0V 0N 1N 1N 60N 120N)

. tran 10ns 140ns
. probe
```

```
. MODEL NMOS NMOS (LEVEL  = 3
+ TOX    = 3.05E-8 NSUB   = 1E15 GAMMA  = 0.7292975
+ PHI    = 0.7 VTO    = 0.6052099 DELTA  = 0.4903542
+ UO     = 800 ETA    = 9.999921E-4 THETA  = 0.0710077
+ KP     = 7.401542E-5 VMAX   = 3.503416E5 KAPPA  = 1.0708
+ RSH    = 0.0299117 NFS    = 4.770051E11 TPG    = 1
+ XJ     = 3E-7 LD     = 7.15092E-12 WD     = 6.615428E-7
+ CGDO   = 1.76E-10 CGSO   = 1.76E-10 CGBO   = 1E-10
+ CJ     = 2.821829E-4 PB     = 0.8883154 MJ     = 0.5
+ CJSW   = 1.244664E-10 MJSW   = 0.05)

. MODEL PMOS PMOS (LEVEL  = 3
+ TOX    = 3.05E-8 NSUB   = 5.327947E16 GAMMA  = 0.4898253
+ PHI    = 0.7 VTO    = -0.8651299 DELTA  = 0.3307574
+ UO     = 159.6662211 ETA    = 1 THETA  = 0.1329612
+ KP     = 2.470055E-5 VMAX   = 3.702724E5 KAPPA  = 10
+ RSH    = 27.1878946 NFS    = 5.440123E11 TPG    = -1
+ XJ     = 2E-7 LD     = 1E-12 WD     = 9.994094E-7
+ CGDO   = 2.25E-10 CGSO   = 2.25E-10 CGBO   = 1E-10
+ CJ     = 3.022339E-4 PB     = 0.7508359 MJ     = 0.4334195
+ CJSW   = 1.781647E-10 MJSW   = 0.1286211)


M19 2 5 7 7 NMOS L=1.6u W=7.2u AD=34.56p PD=24u AS=34.56p PS=24u
* M19 DRAIN GATE SOURCE BULK (10 47 14 65)
M13 2 5 1 1 PMOS L=1.6u W=10.4u AD=49.92p PD=30.4u AS=49.92p PS=30.4u
* M13 DRAIN GATE SOURCE BULK (10 -7 14 19)
C8 5 3 499.37024f
* C8 PLUS MINUS (-240 -23 -47 15)
C7 5 4 499.37024f
* C7 PLUS MINUS (-236 39 -43 77)
C2 5 6 499.37024f
* C2 PLUS MINUS (-239 -151 -46 -113)
C1 5 8 499.37024f
* C1 PLUS MINUS (-240 -87 -47 -49)

. END
```

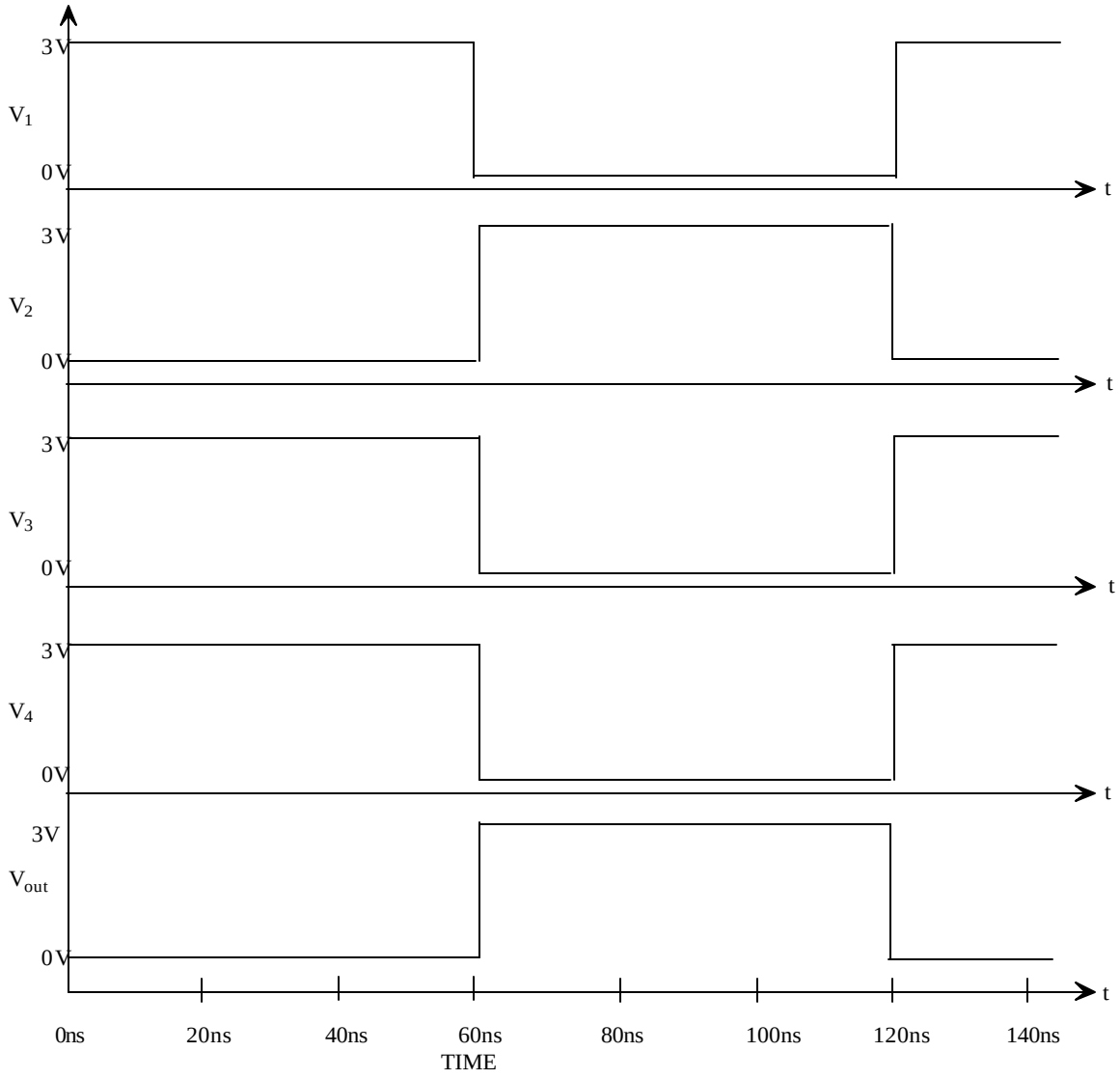


Figure C.2: Input and output waveforms of a 4-input floating gate CMOS inverter of the circuit of Fig. C.1.

Vita

Chandra Srinivasan is from Bangalore, known as ‘silicon valley of India’, in the southern part of India. She has completed her undergraduation from Mysore University in Electrical Engineering in 1997. She joined Louisiana State University, Baton Rouge, in 2000, to pursue a master’s degree in electrical engineering. Her research interests include multivalued logic, multiple-input floating gate MOSFETs and capacitive based architectures. Her other interests include reading, music and international cooking.